

**PENGEMBANGAN APLIKASI MANAJEMEN PERENCANAAN
BAHAN BAKU DAN PEMESANAN PADA KONVEKSI SABLON
MALANG**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

FAKHRUZZAHID WAHDAH

NIM: 145150207111146



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

PENGESAHAN

Pengembangan Aplikasi Manajemen Perencanaan Bahan Baku Dan Pemesanan
Pada Konveksi Sablon Malang

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Fakhruzzahid Wahdah
NIM: 145150207111146

Skripsi ini telah diuji dan dinyatakan lulus pada
30 Juli 2018
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing 2

Bayu Priyambadha, S. Kom, M. Kom
NIP: 19820909 200812 1 004

Denny Sagita Rusdianto, S. Kom, M. Kom
NIP: 19851124 201503 1 001

Mengetahui
Ketua Jurusan Teknik Informatika



Tri Astom Kurniawan, S. T. M. T. Ph. D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiaris, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 17 Juli 2018



Fakhruzzahid Wahdah

NIM: 145150207111146

KATA PENGANTAR

Segala puji bagi Allah SWT atas limpahan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “PENGEMBANGAN APLIKASI MANAJEMEN PERENCANAAN BAHAN BAKU DAN PEMESANAN PADA KONVEKSI SABLON MALANG” .

Untuk kesempatan ini penulis juga menyampaikan rasa terima kasih kepada pihak – pihak yang telah membantu penulis selama penyusunan skripsi, diantaranya:

1. Allah SWT yang telah memberi kemudahan dalam semua proses penulisan skripsi ini.
2. Kepada kedua orang tua penulis, yaitu Bapak Wahyudi dan Ibu Ifadah beserta keluarga besar yang selalu memberikan segala masukan, do’a, motivasi dan semangat yang tidak terputus.
3. Bapak Bayu Priyambadha, S. Kom, M. Kom selaku dosen pembimbing I yang telah meluangkan waktu untuk memberikan masukan, ilmu, serta saran yang bermanfaat dalam proses penyelesaian skripsi ini.
4. Denny Sagita Rusdianto, S.Kom, M.Kom selaku dosen pembimbing II yang juga telah meluangkan waktu untuk memberikan masukan, ilmu, serta saran yang bermanfaat dalam proses penyelesaian skripsi ini.
5. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D, Bapak Ir. Heru Nurwasito, M.Kom, Bapak Drs. Mardji, M.T, dan Bapak Edy Santoso, S.Si, M.Kom selaku Dekan, Wakil Dekan 1, Wakil Dekan 2, dan Wakil Dekan 3 Fakultas Ilmu Komputer, Universitas Brawijaya.
6. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D dan Bapak Bayu Priyambadha, S. Kom, M. Kom selaku Ketua Jurusan Teknik Informatika dan Kepala Program Studi Teknik Informatika Fakultas Ilmu Komputer, Universitas Brawijaya.
7. Seluruh Dosen Fakultas Ilmu Komputer, Universitas Brawijaya atas kesediaannya dalam mengajarkan dan membagikan ilmu yang bermanfaat bagi penulis.
8. Rekan-rekan satu bimbingan yang telah menemani dan membantu saya dalam melakukan bimbingan.
9. Semua pihak yang tidak semuanya bisa dituliskan disini yang terlibat secara langsung maupun tidak langsung dalam proses pengerjaan skripsi maupun sebagai pemberi semangat dan motivasi.

Penulis menyadari masih banyak kekurangan dalam penyusunan skripsi ini baik dalam teknik penyajian materi maupun pembahasan. Demi kesempurnaan

penelitian skripsi ini, saran dan kritik yang sifatnya membangun sangat penulis harapkan. Semoga karya tulis ini bermanfaat dan dapat memberikan sumbangan yang berarti bagi pihak yang membutuhkan.

Malang, 16 Juli 2018

Penulis

fakhruzzahidw@gmail.com



ABSTRAK

Fakhruzzahid Wahdah, Pengembangan Aplikasi Manajemen Perencanaan Bahan Baku Dan Pemesanan Pada Konveksi Sablon Malang

Dosen Pembimbing: Bayu Priyambadha, S. Kom, M. Kom, Denny Sagita Rusdianto, S.Kom, M.Kom

Sablon Malang merupakan salah satu contoh usaha konveksi dan sablon yang beroperasi setiap hari 24 jam, terlebih lagi sablon malang membagi letak produksi terpisah satu sama lain. Tempat toko sablon malang, produksi konveksi, dan produksi sablon terpisah sehingga proses pembuatan pesanan membutuhkan waktu dan tenaga bagi pekerjanya. Sablon malang dalam pencatatan bahan baku juga masih menggunakan manual sehingga dalam proses pengerjaan pesanan sering terjadi kehabisan bahan baku yang mengakibatkan pesanan tersebut menjadi terlambat. Mengatasi permasalahan tersebut dibuatlah sistem manajemen bahan baku dan pemesanan yang diharapkan dapat memudahkan dan memonitoring semua kegiatan proses produksi. Sistem ini menggunakan metode *Material Requirment Planning* untuk memanajemen semua proses pengerjaan baik bahan baku maupun proses pemesanan. Penelitian ini dimulai dari rekayasa kebutuhan sistem yang dilakukan dengan cara wawancara dan observasi langsung terhadap sablon malang. Kemudian dilakukan tahap perancangan yang didalamnya terdapat perancangan arsitektur, perancangan antar muka, perancangan struktur data dan perancangan komponen sedangkan tahap implementasi akan mengimplementasikan semua perancangan yang sudah dibuat, keduanya dilakukan dengan pendekatan *Object Oriented*. Tahap pengujian menggunakan Pengujian Unit, Pengujian Non Fungsional, dan Pengujian Validasi. Hasil dari pengujian persentase valid yaitu 100%.

Kata kunci: konveksi, sablon malang, MRP, manajemen, bahan baku, pemesanan.

ABSTRACT

Fakhruzzahid Wahdah, Development Application Management Planning Raw Material and Booking in Convection Sablon Malang

Supervisor: Bayu Priyambadha, S. Kom, M. Kom, Denny Sagita Rusdianto, S.Kom, M.Kom

Sablon Malang is one of the examples of convection and screen printing business operates 24 hours per day, even more sablon malang split production layout apart from each other. Place the store sablon malang, production of the convection, and the production of screen printing process so that separate orders require time and effort for the workers. Sablon Malang in the record-keeping of raw materials also still using manuals so that the order in the machining process occurs frequently run out of raw materials which resulted in an order to be late. Overcoming these problems made the raw material management system and bookings are expected to facilitate and monitor all activities of the production process. This system uses the method Requirement Material Planning for the create all machining process both raw materials as well as the booking process. This research starts from the needs analysis system that is carried out by means of direct observation and interviews in Sablon Malang. Then done stage design which contained the design architecture, designing interfaces, designing data structures and designing components while the implementation phase will implement all of the design already made, both performed with the Object Oriented approach. Testing phase using Unit testing, Non Functional Testing, and Validation Testing. The results of testing valid percentage i.e. 100%.

Keywords: convection, sablon, MRP, management, raw materials, booking.

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xvii
DAFTAR LAMPIRAN	xx
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan.....	2
1.4 Manfaat	2
1.5 Batasan Masalah	3
1.6 Sistematika Penulisan	3
BAB 2 LANDASAN KEPUSTAKAAN	4
2.1 Kajian Pustaka	4
2.2 Pengertian Bahan Baku	5
2.3 Pengembangan Perangkat Lunak.....	5
2.3.1 Model Pengembangan perangkat Lunak	6
2.3.2 Pendekatan Object Oriented	10
2.4 Teknologi Pengembangan Sistem	16
2.4.1 <i>Material Requirement Planning</i>	16
2.4.2 <i>PHP Hypertext Preprocessor (PHP)</i>	18
2.4.3 MySQL	19
2.4.4 <i>CodeIgniter</i>	19
BAB 3 METODOLOGI	20
3.1 Studi Literatur	21
3.2 Rekayasa Kebutuhan.....	21
3.3 Perancangan Sistem.....	22

3.4 Implementasi.....	23
3.5 Pengujian dan Analaisis.....	23
3.6 Kesimpulan	24
BAB 4 REKAYASA KEBUTUHAN.....	25
4.1 Elisitasi Kebutuhan	25
4.1.1 Gambaran Umum Sistem	26
4.2 Spesifikasi Kebutuhan	26
4.2.1 Identifikasi Aktor	26
4.2.2 Analisis Kebutuhan Fungsional	27
4.2.3 Analisis Kebutuhan Non Fungsional.....	31
4.2.4 Use Case Diagram	32
4.2.5 Use Case Scenario	33
4.3 <i>Entity Relationship Diagram (ERD)</i>	54
BAB 5 PERANCANGAN DAN IMPLEMENTASI	56
5.1 Perancangan.....	56
5.1.1 Perancangan Arsitektur.....	56
5.1.2 Perancangan Struktur Data.....	61
5.1.3 Perancangan Komponen	62
5.1.4 Perancangan Antarmuka.....	65
5.2 Implementasi.....	93
5.2.1 Spesifikasi Sistem	93
5.2.2 Batasan Implementasi.....	94
5.2.3 Implementasi Basis Data	94
5.2.4 Implementasi <i>Class</i>	95
5.2.5 Implementasi Kode Program	96
5.2.6 Implementasi Antarmuka	100
BAB 6 PENGUJIAN	115
6.1 Pengujian Unit.....	115
6.1.1 Pengujian Unit Method ProsesCek pada klas CatatanBaru	115
6.1.2 Pengujian Unit Method prosesTambah pada klas RequestOrder .	120
6.1.3 Pengujian Unit Method prosesTambah pada klas JadwalProduksiBaru	122
6.2 Pengujian Validasi	125

6.2.1 Pengujian Login	125
6.2.2 Pengujian Logout.....	127
6.2.3 Pengujian Membuat Pesanan	127
6.2.4 Pengujian Melihat Pesanan.....	129
6.2.5 Pengujian Berhasil Menghapus Pesanan	130
6.2.6 Pengujian Mengubah Pesanan.....	130
6.2.7 Pengujian Mengunggah Gambar Pesanan	131
6.2.8 Pengujian Melihat Bahan Konveksi.....	133
6.2.9 Pengujian Melihat Bahan Sablon	133
6.2.10 Pengujian Menambahkan Bahan Baku	134
6.2.11 Pengujian Melihat Bahan Konveksi dan Sablon.....	135
6.2.12 Pengujian Menghapus Bahan Konveksi dan Sablon	135
6.2.13 Pengujian Mengubah Bahan Konveksi dan Sablon.....	136
6.2.14 Pengujian Mengunggah Gambar Bahan Baku	137
6.2.15 Pengujian Membuat Pembelian Bahan Baku.....	138
6.2.16 Pengujian Mengubah Pembelian Bahan Baku	139
6.2.17 Pengujian Melihat Pembelian Bahan Baku	140
6.2.18 Pengujian Menghapus Pembelian Bahan Baku	141
6.2.19 Pengujian Membuat Jadwal Induk Produksi.....	141
6.2.20 Pengujian Melihat Jadwal Induk Produksi	142
6.2.21 Pengujian Mengubah Jadwal Induk Produksi	143
6.2.22 Pengujian Menghapus Jadwal Induk Produksi.....	144
6.2.23 Pengujian Membuat Request Order	144
6.2.24 Pengujian Mengubah Request Order.....	145
6.2.25 Pengujian Melihat Request Order.....	147
6.2.26 Pengujian Menghapus Request Order	147
6.2.27 Pengujian Memvalidasi Request Order	147
6.2.28 Pengujian Mengecek Ketersediaan.....	148
6.2.29 Pengujian Update Status Pesanan	150
6.2.30 Pengujian Mencetak Nota.....	150
6.2.31 Pengujian Membuat Catatan Pesanan.....	151
6.2.32 Pengujian Melihat Catatan.....	152
6.2.33 Pengujian Menambahkan User	153

6.2.34 Pengujian Melihat Daftar User.....	154
6.2.35 Pengujian Mengubah User.....	154
6.2.36 Pengujian Berhasil Menghapus User	155
6.3 Pengujian Non Fungsional.....	156
6.4 Pembahasan Hasil Pengujian	156
BAB 7 PENUTUP	157
7.1 Kesimpulan.....	157
7.2 Saran.....	157
DAFTAR PUSTAKA.....	159
LAMPIRAN A BISNIS PROSES SABLON MALANG	160



DAFTAR TABEL

Tabel 2.1 Komponen Use Case Diagram	12
Tabel 2.2 Sequence Diagram.....	14
Tabel 2.3 Relasi Class Diagram	15
Tabel 4.1. Tabel Aktor dan Penjelasanya	27
Tabel 4.2 Tabel Kebutuhan Fungsional	27
Tabel 4.3 Tabel Kebutuhan Non Fungsional	31
Table 4.4 Skenario Use Case Login.....	33
Table 4.5 Skenario Use Case Logout	34
Table 4.6 Skenario Use Case Membuat Pesanan	34
Table 4.7 Skenario Use Case Mengunggah Gambar Pesanan.....	35
Table 4.8 Skenario Use Case Mencetak Nota	35
Table 4.9 Skenario Use Case Mengubah Pesanan	36
Table 4.10 Skenario Use Case Melihat Pesanan	36
Table 4.11 Skenario Use Case Melihat Jadwal Induk Produksi	37
Table 4.12 Skenario Use Case Update Status Pesanan	37
Table 4.13 Skenario Use Case Mencatat Hasil Pesanan	38
Table 4.14 Skenario Use Case Melihat Catatan Pesanan.....	39
Table 4.15 Skenario Use Case Membuat Jadwal Induk Produksi	40
Table 4.16 Skenario Use Case Mengubah Jadwal Induk Produksi	40
Table 4.17 Skenario Use Case Mengecek Ketersediaan	41
Table 4.18 Skenario Use Case Membuat Request Order	42
Table 4.19 Skenario Use Case Mengubah Request Order	43
Table 4.20 Skenario Use Case Melihat Request Order	44
Table 4.21 Skenario Use Case Melihat Bahan Konveksi	44
Table 4.22 Skenario Use Case Melihat Bahan Konveksi	45
Table 4.23 Skenario Use Case Memvalidasi Request Order	45
Table 4.24 Skenario Use Case Menambahkan Bahan Baku.....	46
Table 4.25 Skenario Use Case Melihat Bahan Konveksi dan Sablon	46
Table 4.26 Skenario Use Case Mengubah Bahan Sablon dan Konveksi	47
Table 4.27 Skenario Use Case Mengunggah Gambar Bahan Baku	47
Table 4.28 Skenario Use Case Membuat Pembelian Bahan Baku	48

Table 4.29 Skenario Use Case Melihat Pembelian Bahan Baku.....	49
Table 4.30 Skenario Use Case Mengubah Pembelian Bahan Baku.....	49
Table 4.31 Skenario Use Case Menghapus Pesanan.....	50
Table 4.32 Skenario Use Case Menghapus Bahan Konveksi dan Sablon.....	50
Table 4.33 Skenario Use Case Menghapus Pembelian Bahan baku	51
Table 4.34 Skenario Use Case Menghapus Jadwal Induk Produksi	51
Table 4.35 Skenario Use Case Menghapus Request Order.....	52
Table 4.36 Skenario Use Case Menambah User	52
Table 4.37 Skenario Use Case Mengubah User	53
Table 4.38 Skenario Use Case Melihat User	53
Table 4.39 Skenario Use Case Menghapus User.....	54
Table 5.1 Penjelasan Antarmuka Halaman Login	66
Table 5.2 Penjelasan Antarmuka Halaman Tambah Pesanan	67
Table 5.3 Penjelasan Antarmuka Halaman Daftar Pesanan	68
Table 5.4 Penjelasan Antarmuka Halaman Ubah Pesanan.....	69
Table 5.5 Penjelasan Antarmuka Halaman Ubah Pesanan.....	70
Table 5.6 Penjelasan Antarmuka Halaman Jadwal Produksi Pesanan.....	71
Table 5.7 Penjelasan Antarmuka Halaman Check Persediaan.....	72
Table 5.8 Penjelasan Antarmuka Halaman Tambah Catatan	73
Table 5.9 Penjelasan Antarmuka Halaman Daftar Catatan Pesanan.....	74
Table 5.10 Penjelasan Antarmuka Halaman Tambah Bahan Baku	75
Table 5.11 Penjelasan Antarmuka Halaman Daftar Bahan Baku	76
Table 5.12 Penjelasan Antarmuka Halaman Detail Bahan Baku.....	77
Table 5.13 Penjelasan Antarmuka Halaman Daftar Persediaan Bahan Baku	78
Table 5.14 Penjelasan Antarmuka Halaman Tambah Request Order	79
Table 5.15 Penjelasan Antarmuka Halaman Daftar Request Order	80
Table 5.16 Penjelasan Antarmuka Halaman Detail Request Order	81
Table 5.17 Penjelasan Antarmuka Halaman Validasi Request Order	82
Table 5.18 Penjelasan Antarmuka Halaman Tambah Jadwal Induk Produksi	83
Table 5.19 Penjelasan Antarmuka Halaman Daftar Jadwal Induk Produksi	84
Table 5.20 Penjelasan Antarmuka Halaman Detail Jadwal Induk Produksi.....	85
Table 5.21 Penjelasan Antarmuka Halaman Tambah Pembelian.....	86

Table 5.22 Penjelasan Antarmuka Halaman Daftar Pembelian	87
Table 5.23 Penjelasan Antarmuka Halaman Detail Pembelian	89
Table 5.24 Penjelasan Antarmuka Halaman Tambah User.....	90
Table 5.25 Penjelasan Antarmuka Halaman Daftar User	90
Table 5.26 Penjelasan Antarmuka Halaman Detail User	91
Table 5.27 Penjelasan Antarmuka Halaman Mencetak Nota	92
Tabel 5.28 Spesifikasi Perangkat Keras	93
Tabel 5.29 Spesifikasi Perangkat Lunak	93
Tabel 5.30 Implementasi Class Diagram	95
Tabel 6.1 Hasil Pengujian Unit method prosesCheck pada klas CatatanBaru	118
Tabel 6.2 Hasil Pengujian Unit method prosesTambah pada klas RequestOrder	121
Tabel 6.3 Hasil Pengujian Unit method prosesTambah pada klas JadwalProduksiBaru	123
Tabel 6.4 Kasus Uji Berhasil Login Sebagai Kasir	125
Tabel 6.5 Kasus Uji Berhasil Login Sebagai Gudang	125
Tabel 6.6 Kasus Uji Berhasil Login Sebagai Sablon	126
Tabel 6.7 Kasus Uji Berhasil Login Sebagai Konveksi	126
Tabel 6.8 Kasus Uji Data Login Kosong	126
Tabel 6.9 Kasus Uji Data Login Salah	127
Tabel 6.10 Kasus Uji Logout	127
Tabel 6.11 Kasus Uji Berhasil Membuat Pesanan	127
Tabel 6.12 Kasus Uji Form Pesanan Kosong	128
Tabel 6.13 Kasus Uji Form Jumlah Pesanan Melebihi Kapasitas Produksi	128
Tabel 6.14 Kasus Uji Melihat Pesanan	129
Tabel 6.15 Kasus Uji Berhasil Menghapus Pesanan	130
Tabel 6.16 Kasus Uji Berhasil Mengubah Pesanan	130
Tabel 6.17 Kasus Uji Form Pesanan Kosong	131
Tabel 6.18 Kasus Uji Berhasil Mengunggah Gambar Pesanan	131
Tabel 6.19 Kasus Uji Mengunggah Gambar tidak Sesuai Format	132
Tabel 6.20 Kasus Uji Melihat Bahan Konveksi	133
Tabel 6.21 Kasus Uji Melihat Bahan Sablon	133

Tabel 6.22 Kasus Uji Berhasil Menambahkan Bahan Baku	134
Tabel 6.23 Kasus Uji Form Bahan Baku Baru Kosong.....	134
Tabel 6.23 Kasus Uji Melihat Bahan Konveksi dan Sablon.....	135
Tabel 6.24 Kasus Uji Berhasil Menghapus Bahan Konveksi dan Sablon	135
Tabel 6.25 Kasus Uji Berhasil Mengubah Bahan Konveksi dan Sablon.....	136
Tabel 6.26 Kasus Uji Form Bahan Baku Kosong	136
Tabel 6.27 Kasus Uji Berhasil Mengunggah Gambar Bahan Baku	137
Tabel 6.28 Kasus Uji Mengunggah Gambar tidak Sesuai Format	138
Tabel 6.29 Kasus Uji Berhasil Membuat Pembelian Bahan Baku	138
Tabel 6.30 Kasus Uji Form Pembelian Kosong	139
Tabel 6.31 Kasus Uji Berhasil Mengubah Pembelian Bahan Baku	139
Tabel 6.32 Kasus Uji Form Pembelian Kosong	140
Tabel 6.33 Kasus Uji Melihat Pembelian Bahan Baku.....	140
Tabel 6.34 Kasus Uji Berhasil Menghapus Pembelian Bahan Baku	141
Tabel 6.35 Kasus Uji Berhasil Membuat Jadwal Induk Produksi.....	141
Tabel 6.36 Kasus Uji Form Jadwal Produksi Baru Kosong.....	142
Tabel 6.37 Kasus Uji Melihat Jadwal Induk Produksi	142
Tabel 6.38 Kasus Uji Berhasil Mengubah Jadwal Induk Produksi	143
Tabel 6.39 Kasus Uji Form Jadwal Produksi Kosong	143
Tabel 6.40 Kasus Uji Berhasil Menghapus Jadwal Induk Produksi	144
Tabel 6.41 Kasus Uji Berhasil Membuat Request Order	144
Tabel 6.42 Kasus Uji Form Jadwal Produksi Baru Kosong.....	145
Tabel 6.43 Kasus Uji Berhasil Mengubah Request Order	145
Tabel 6.44 Kasus Uji Form Request Order Kosong.....	146
Tabel 6.45 Kasus Uji Melihat Request Order	147
Tabel 6.46 Kasus Uji Berhasil Menghapus Request Order	147
Tabel 6.47 Kasus Uji Memvalidasi Disetujui Request Order	148
Tabel 6.48 Kasus Uji Form Memvalidasi Tolak Request Order	148
Tabel 6.49 Kasus Uji Berhasil Mengecek Ketersediaan	149
Tabel 6.50 Kasus Uji Form Bahan Tidak Tersedia	149
Tabel 6.51 Kasus Uji Update Status Pesanan	150
Tabel 6.52 Kasus Uji Update Mencetak Nota	150

Tabel 6.53 Kasus Uji Berhasil Membuat Catatan Pesanan	151
Tabel 6.54 Kasus Uji Form Catatan Pesanan Kosong	152
Tabel 6.55 Kasus Uji Melihat Catatan	152
Tabel 6.56 Kasus Uji Berhasil Menambahkan User.....	153
Tabel 6.57 Kasus Uji Form User Baru Kosong	153
Tabel 6.58 Kasus Uji Melihat Daftar User	154
Tabel 6.59 Kasus Uji Berhasil Mengubah User.....	154
Tabel 6.60 Kasus Uji Form User Kosong.....	155
Tabel 6.61 Kasus Uji Berhasil Menghapus User	155



DAFTAR GAMBAR

Gambar 2.1 Tahapan <i>waterfall</i> model	7
Gambar 3.1 Tahapan Metode Penelitian	20
Gambar 4.1 <i>Use Case</i> Sistem.....	32
Gambar 4.2 <i>Entity Relationship Diagram (ERD)</i>	55
Gambar 5.1 <i>Sequence diagram</i> Mengubah Bahan Baku	57
Gambar 5.2 <i>Sequence diagram</i> Menambah Bahan Baku	58
Gambar 5.3 <i>Sequence diagram</i> Mencatat Hasil Pesanan Konveks.....	59
Gambar 5.5 Conceptual <i>Data Model</i>	61
Gambar 5.6 <i>Physical Data Model</i>	62
Gambar 5.7 Tampilan Antarmuka Halaman Login	66
Gambar 5.8 Tampilan Antarmuka Halaman Tambah Pesanan	67
Gambar 5.9 Tampilan Antarmuka Halaman Daftar Pesanan.....	68
Gambar 5.10 Tampilan Antarmuka Halaman Ubah Pesanan	69
Gambar 5.11 Tampilan Antarmuka Halaman Ubah Pesanan	70
Gambar 5.12 Tampilan Antarmuka Halaman Jadwal Produksi Pesanan	71
Gambar 5.13 Tampilan Antarmuka Halaman Check Persediaan	72
Gambar 5.14 Tampilan Antarmuka Halaman Tambah Catatan.....	73
Gambar 5.15 Tampilan Antarmuka Halaman Daftar Catatan Pesanan	74
Gambar 5.16 Tampilan Antarmuka Halaman Tambah Bahan Baku	75
Gambar 5.17 Tampilan Antarmuka Halaman Daftar Bahan Baku	76
Gambar 5.18 Tampilan Antarmuka Halaman Detail Bahan Baku	77
Gambar 5.19 Tampilan Antarmuka Halaman Daftar Persediaan Bahan Baku	78
Gambar 5.20 Tampilan Antarmuka Halaman Tambah Request Order.....	79
Gambar 5.21 Tampilan Antarmuka Halaman Daftar Request Order.....	80
Gambar 5.22 Tampilan Antarmuka Halaman Detail Request Order	81
Gambar 5.23 Tampilan Antarmuka Halaman Validasi Request Order	82
Gambar 5.24 Tampilan Antarmuka Halaman Tambah Jadwal Induk Produksi	83
Gambar 5.25 Tampilan Antarmuka Halaman Daftar Jadwal Induk Produksi	84
Gambar 5.26 Tampilan Antarmuka Halaman Detail Jadwal Induk Produksi	85

Gambar 5.27 Tampilan Antarmuka Halaman Tambah Pembelian	86
Gambar 5.28 Tampilan Antarmuka Halaman Daftar Pembelian	87
Gambar 5.29 Tampilan Antarmuka Halaman Detail Pembelian.....	88
Gambar 5.30 Tampilan Antarmuka Halaman Tambah User	89
Gambar 5.31 Tampilan Antarmuka Halaman Daftar User	90
Gambar 5.32 Tampilan Antarmuka Halaman Detail User.....	91
Gambar 5.33 Tampilan Antarmuka Halaman Mencetak Nota	92
Gambar 5.34 Implementasi Database	94
Gambar 5.35 Implementasi Halaman Login.....	101
Gambar 5.36 Implementasi Halaman Tambah Pesanan.....	101
Gambar 5.37 Implementasi Halaman Daftar Pesanan	102
Gambar 5.38 Implementasi Halaman Detail Pesanan	102
Gambar 5.39 Implementasi Halaman Ubah Pesanan	103
Gambar 5.40 Implementasi Halaman Jadwal Produksi Pesanan.....	104
Gambar 5.41 Implementasi Halaman Check Persediaan.....	104
Gambar 5.42 Implementasi Halaman Tambah Catatan.....	105
Gambar 5.43 Implementasi Halaman Daftar Catatan	105
Gambar 5.44 Implementasi Halaman Tambah Bahan baku	106
Gambar 5.46 Implementasi Halaman Daftar Bahan Baku	106
Gambar 5.46 Implementasi Halaman Detail Bahan Baku.....	107
Gambar 5.47 Implementasi Halaman Daftar Persediaan Bahan baku	107
Gambar 5.48 Implementasi Halaman Request Order.....	108
Gambar 5.49 Implementasi Halaman Daftar Request Order	108
Gambar 5.50 Implementasi Halaman Detail Request Order	109
Gambar 5.51 Implementasi Halaman Validasi Request Order	109
Gambar 5.52 Implementasi Halaman Tambah Jadwal Produksi	110
Gambar 5.53 Implementasi Halaman Daftar Jadwal Produksi	110
Gambar 5.54 Implementasi Halaman Detail Jadwal Produksi.....	111
Gambar 5.55 Implementasi Halaman Tambah Pembelian	111
Gambar 5.56 Implementasi Halaman Daftar Pembelian.....	112
Gambar 5.57 Implementasi Halaman Detail Pembelian.....	112
Gambar 5.58 Implementasi Halaman Tambah User.....	113

Gambar 5.59 Implementasi Halaman Daftar User.....	113
Gambar 5.60 Implementasi Halaman Detail User	114
Gambar 5.61 Implementasi Halaman Nota	114



DAFTAR LAMPIRAN

Lampiran A Proses Bisnis Sablon Malang	160
--	-----



BAB 1 PENDAHULUAN

1.1 Latar belakang

Potensi Indonesia dalam bidang konveksi dan sablon didunia masih besar karena dalam hal fasilitas diindustri lengkap dan memiliki tenaga kerja usia produktif yang melimpah. Hal ini terbukti dengan terus meningkatnya nilai ekspor tekstil dari tahun ke tahun. Pada tahun 2010 ekspor tekstil senilai US\$ 5.616.916.646 mengalami peningkatan sebesar 14,058% dari nilai ekspor tekstil tahun sebelumnya, yaitu senilai US\$ 4.924.635.472 . Saat tahun 2011, ekspor tekstil Indonesia telah mencapai US\$ 6.656.898.609. Pencapaian ini memberikan kontribusi sebesar 5,45% terhadap total ekspor hasil industri di Indonesia (www.kemenperin.go.id).

Konveksi sendiri adalah tempat produksi *marchandise* yang berhubungan dengan kebutuhan sandang atau pakaian, atau lebih modernnya berhubungan dengan produksi *fashion* dengan minimal pemesanan tertentu. Konveksi biasanya membuka jasa pembuatan baju kaos dan sablon satuan, grup, komunitas. Bisa juga berupa seragam, celana training, jaket, polo shirt, tas kanvas dan lainnya yang berhubungan dengan jahit menjahit dan sablon, baik sablon manual maupun digital. Ramainya permintaan pasar dan besarnya prospek bisnis yang dijanjikan membuat sebagian orang menjadi tertarik menekuni usaha konveksi dan sablon, baik itu sablon manual maupun sablon digital.

Sablon Malang merupakan salah satu contoh usaha konveksi dan sablon yang terletak di kabupaten Malang kecamatan Karangploso. Konveksi ini beroperasi setiap hari 24 jam sehingga termasuk usaha yang membutuhkan waktu dan tenaga bagi pekerjanya. Sablon Malang melayani berbagai macam pemesanan konveksi, dari yang memesan dari awal proses yaitu jahit kaos dan sablon atau hanya melakukan sablon saja. Sablon sendiri memiliki berbagai jenis cat yang digunakan seperti *Rubber*, *Plastisol* atau *Discars*. Konveksi jahit juga memiliki berbagai jenis kain yang disediakan mulai dari *Cotton Combed* 20s, 30s, 40s dan ada juga jenis kain *Bamboo*. Pada proses pemesanan di Sablon Malang terdiri dari beberapa tahap untuk menjadi hasil akhir produk. Pertama, pelanggan datang ke kantor Sablon Malang kemudian mengisi data dan mengirimkan berkas pesanan. Kedua, konveksi menerima berkas dan mengerjakan proses jahit sesuai pesanan. Ketiga, ketika konveksi sudah menyelesaikan proses jahit akan di kirim ke tempat sablon kemudian dilakukan proses sablon sesuai dengan pesanan. Terakhir, jika sablon sudah selesai maka akan dikemas sesuai dengan pesanan dan akan diantarkan ke kantor untuk diambil kembali kepada pelanggan

Banyaknya bahan baku yang harus dipenuhi membuat pemilik harus mengecek bahan baku masih tersedia, tetapi dalam pelaksanaanya Sablon Malang jarang mengecek bahan baku tersebut sehingga mengakibatkan habisnya bahan baku pada saat proses pembuatan pesanan. Menurut pemilik sablon malang pencatatan bahan baku masih menggunakan cara manual tidak dengan bantuan komputer, menyebabkan sablon malang tidak memprediksi persediaan bahan baku untuk memenuhi pesanan yang baru. Dampak dari hal tersebut adalah bahan baku tidak mencukupi tetapi masih dalam proses pembuatan pesanan, sehingga dampak selanjutnya adalah akan melakukan pembelian ke tempat lain yang menyebabkan terbuangnya waktu proses pemesanan. Ketika melakukan proses

pemesanan pegawai Sablon Malang sering kali terlambat dalam menyelesaikannya, dikarenakan beberapa faktor salah satunya yaitu karena dalam proses pembuatan pesanan tidak dipantau langsung pemilik dan pimpinan produksi sehingga monitoring dalam proses tersebut masih belum baik. Dalam proses pemesanan sablon malang hanya memakai catatan manual yaitu nota, serta sablon malang tidak mempunyai data pelanggan yang sebelumnya sudah pernah membuat pesanan.

Mengatasi permasalahan tersebut diperlukan sistem manajemen bahan baku dan pemesanan yang diharapkan dapat membantu seluruh kegiatan dalam melakukan proses pemesanan di Sablon Malang sehingga tidak terjadi lagi kehabisan bahan baku pada saat proses pembuatan pesanan dan keterlambatan pesanan sesuai persetujuan pelanggan. Sistem ini akan dibuat menggunakan *platform web*, karena web merupakan media yang dapat diakses secara bebas baik waktu maupun tempat. Sistem ini akan dibangun menggunakan bahasa pemrograman PHP dan *CodeIgniter*. Dimana *CodeIgniter* sendiri merupakan salah satu *framework* tercepat dibandingkan dengan *framework* lainnya (Daqiqil, 2011). Sistem manajemen bahan baku dan pesanan tersebut akan dibuat dalam penelitian skripsi dengan judul “Pengembangan Aplikasi Manajemen Perencanaan Bahan Baku dan Pemesanan pada Konveksi Sablon Malang”.

1.2 Rumusan Masalah

Berdasarkan pada latar belakang yang telah diuraikan, maka didapatkan rumusan masalah sebagai berikut :

1. Bagaimana hasil rekayasa kebutuhan sistem aplikasi manajemen bahan baku dan pemesanan yang sesuai dengan proses bisnis yang terjadi pada sablon malang?
2. Bagaimana hasil perancangan dan implementasi aplikasi manajemen perencanaan bahan baku dan pemesanan pada Sablon Malang berdasarkan rekayasa kebutuhan yang telah dilakukan sebelumnya?
3. Apakah hasil pengujian aplikasi manajemen perencanaan bahan baku dan pemesanan pada Sablon Malang berhasil menjawab dan memenuhi kebutuhan sistem yang telah dikembangkan?

1.3 Tujuan

Berdasarkan rumusan masalah didapatkan tujuan penelitian ini adalah :

1. Merekayasa Kebutuhan sistem aplikasi yang sesuai berdasarkan proses bisnis yang terjadi pada sablon malang.
2. Merancang dan mengimplementasikan aplikasi manajemen perencanaan bahan baku dan pemesanan pada Sablon Malang.
3. Menganalisa hasil pengujian aplikasi manajemen perencanaan bahan baku dan pemesanan pada Sablon Malang telah memenuhi kebutuhan sistem.

1.4 Manfaat

Manfaat yang diharapkan dapat dicapai dari penelitian ini adalah :

1. Membuat dan Mengembangkan sebuah sistem sesuai dengan kebutuhan yang didapat untuk Sablon Malang.

2. Pengembang selanjutnya dapat mengetahui secara rinci dari sistem.
3. Pemilik dapat memeriksa informasi terkait dengan usaha Seperti Informasi Pelanggan, Informasi Proses Pesanan dan Bahan Baku secara online.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini adalah :

1. Studi kasus penelitian tempat konveksi yaitu Sablon Malang.
2. Lingkup permasalahan hanya berada pada produksi sablon, konveksi dan kasir.
3. Kasus yang diselesaikan hanya masalah pencatatan bahan baku konveksi dan sablon, pencatatan pemesanan, dan monitoring proses pembuatan pesanan konveksi dan sablon.

1.6 Sistematika Penulisan

Untuk mencapai tujuan yang diharapkan, maka sistematika pembahasan yang disusun dalam skripsi adalah sebagai berikut :

1. BAB 1 PENDAHULUAN

Bab ini berisi penjabaran latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika penulisan.

2. BAB 2 LANDASAN KEPUSTAKAAN

Bab ini berisi tentang teori-teori pendukung dan bahan penelitian mengenai teknologi yang diimplementasikan pada penelitian ini.

3. BAB 3 METODOLOGI

Bab ini membahas tentang metode yang akan digunakan dalam proses penulisan yang terdiri dari studi literatur, rekayasa kebutuhan, perancangan sistem, implementasi, pengujian dan analisis, dan pengambilan kesimpulan dan saran.

4. BAB 4 REKAYASA KEBUTUHAN

Bab ini berisikan tentang bagaimana rekayasa kebutuhan yang direpresentasikan dengan menggunakan *use case* dan *use case scenario*.

5. BAB 5 PERANCANGAN DAN IMPLEMENTASI

Bab ini membahas tentang perancangan perangkat lunak berdasarkan analisis kebutuhan dan akan dirancang menggunakan pemrograman HTML, PHP, Javascript, CSS, *sequence diagram* dan *class diagram*. Selanjutnya merancang hal-hal yang saling berhubungan dengan hasil analisis tersebut kemudian mengimplementasikan pada aplikasi.

6. BAB 6 PENGUJIAN DAN ANALISIS

Bab ini memuat tentang metode pengujian dan analisis terhadap sistem yang telah dirancang dan diimplementasikan.

7. BAB 7 KESIMPULAN

Bab ini berisikan kesimpulan yang telah diperoleh dari perancangan, implementasi, dan pengujian perangkat lunak yang dibangun dalam skripsi ini serta beberapa saran untuk pengembangan sistem lebih lanjut.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Penelitian Penelitian terdahulu mengenai penerapan metode MRP (*Material Requirement Planning*) sebagai upaya untuk meningkatkan efisiensi biaya produksi telah banyak dilakukan. Seperti yang dilakukan oleh Sudarmaji (2010) yang berjudul “Perencanaan Persediaan Bahan Baku Menggunakan Metode MRP di PR. Cengkir Gading Nganjuk” yang menyimpulkan bahwa perhitungan biaya bahan baku selama 2009, total biaya yang dikeluarkan dengan menggunakan metode perusahaan sebesar Rp. 5.727.840.000,- lebih besar dari metode MRP yaitu sebesar Rp. 5.606.598.000,- dengan menggunakan metode MRP perusahaan dapat mengurangi biaya hingga Rp. 121.242.000,- atau penghematan sebesar 2,1%. dan interval pemesanan dari dua belas kali menjadi enam kali selama setahun. Perencanaan kebutuhan persediaan bahan baku untuk satu tahun mendatang menggunakan metode MRP adalah bahan baku tembakau sebesar 74758 kg, bahan baku saus 5979 ltr, bahan baku cengkeh 37378 kg, dan bahan baku kertas 2595 dos dengan total biaya pengadaan bahan baku Rp. 5.084.658.000,-.

Harahap (2010) yang berjudul “Analisis Penerapan *Material Requirement Planning* dalam Perencanaan Bahan Pembuatan *Safety Industry Shoes* (Studi Kasus pada Home Industry di CV. VANNY Shoes)” yang menyimpulkan bahwa perencanaan persediaan bahan baku oleh perusahaan selama tiga bulan terakhir tahun 2009 sebanyak dua belas kali kurang teratur dengan biaya Rp. 100.153.800. apabila menerapkan metode MRP, persediaan bahan baku hanya akan dilakukan sebanyak delapan kali dan dilakukan lebih teratur karena terencana dengan biaya Rp. 78.733.441. Persediaan bahan baku dengan menggunakan metode MRP lebih baik karena akan mengurangi biaya hingga Rp. 21.420.359 atau sebesar 21,4% dan lebih optimal dalam memenuhi jumlah persediaan di gudang.

Surianto (2013) yang berjudul “Penerapan Metode *Material Requirement Planning* Di PT. Bokormas Mojokerto” yang menyimpulkan bahwa penerapan MRP memberikan manfaat bagi perusahaan berupa penghematan biaya pengendalian. Penghematan dapat tercapai karena dalam sistem MRP menekankan tingkat persediaan bahan baku seminimal mungkin sesuai dengan kebutuhan. Dengan keputusan melakukan produksi sesuai dengan jumlah permintaan, perusahaan dapat menghemat biaya pengendalian persediaan bahan baku sebesar Rp 979.659.983 atau sebesar 72% dari keadaan sebelumnya sebesar Rp 1.359.299.820 sedangkan penerapan MRP menghasilkan total biaya pengendalian persediaan bahan baku sebesar Rp Rp 379.639.837 dengan jumlah produksi sesuai dengan jumlah permintaan. Perusahaan lebih efisien dengan melakukan produksi sesuai dengan permintaan karena tidak ada biaya penyimpanan sebesar Rp 183.535 yang timbul karena kelebihan produksi.

Ummiroh (2013) yang berjudul “Analisis Penerapan MRP pada Pennyellow Furniture” yang menyimpulkan bahwa analisis *Material Requirement Planning* (MRP) pada Pennyellow Furniture dilakukan secara manual, karena jumlah komponen yang terlihat dalam produksi relatif sedikit. *Material Requirement Planning* (Rencana Kebutuhan Bahan) pada Pennyellow Furniture adalah sebagai berikut: bahan baku berupa rotan sintesis, pipa

alumunium, dan aksesoris metal dipesan setiap tiga minggu sekali dan tersedia di gudang satu minggu kemudian, pembuatan kursi dan meja langsung dilaksanakan ketika bahan baku sampai di gudang dan siap untuk dilakukan proses penyelesaian satu minggu kemudian, dan produk jadi dapat selesai satu minggu dari dimulainya proses penyelesaian. Jadi, pemesanan bahan baku, proses produksi hingga selesainya produk akhir adalah selama tiga minggu.

2.2 Pengertian Bahan Baku

Adapun pengertian bahan baku menurut Farah Margaret (2007:147) adalah “Persediaan bahan baku merupakan bahan baku atau bahan tambahan yang dimiliki oleh perusahaan untuk digunakan dalam aktifitas proses produksi persediaan material menjadi komponen utama dari suatu produk.”. Menurut Fredy Rangkuti (2007:425) persediaan bahan baku adalah “Persediaan bahan baku mempunyai kedudukan yang penting dalam perusahaan karena persediaan bahan baku sangat besar pengaruhnya terhadap kelancaran produksi.”

Dari definisi diatas dapat disimpulkan bahwa bahan baku merupakan bahan yang utama didalam melakukan proses produksi sampai menjadi barang jadi. Bahan baku meliputi semua barang dan bahan yang dimiliki perusahaan dan digunakan untuk proses produksi (Singgih Wibowo, 2007:24). Menurut (Masiyal Kholmi 2003:172) bahan baku memiliki beberapa faktor yang perlu diperhatikan, yaitu :

1. Perkiraan pemakaian
Merupakan perkiraan tentang jumlah bahan baku yang akan digunakan oleh perusahaan untuk proses produksi pada periode yang akan datang.
2. Harga bahan baku
Merupakan dasar penyusunan perhitungan dari perusahaan yang harus disediakan untuk investasi dalam bahan baku tersebut.
3. Biaya-biaya persediaan
Merupakan biaya-biaya yang dibutuhkan oleh perusahaan untuk pengadaan bahan baku
4. Kebijakan pembelanjaan
Merupakan faktor penentu dalam menentukan berapa besar persediaan bahan baku yang akan mendapatkan dana dari perusahaan.
5. Pemakaian sesungguhnya
Merupakan pemakaian bahan baku yang sesungguhnya dari periode lalu dan merupakan salah satu faktor yang perlu diperhatikan.
6. Waktu tunggu
Merupakan tenggang waktu yang tepat maka perusahaan dapat membeli bahan baku pada saat yang tepat pula, sehingga resiko penumpukan ataupun kekurangan persediaan dapat ditekan seminimal mungkin.

2.3 Pengembangan Perangkat Lunak

Pengembangan perangkat lunak merupakan tahapan dimana memusatkan perhatiannya pada kontruksi data, implementasi fungsi, implementasi prosedur, pembuatan antar muka, bagaimana perancangan akan diterjemahkan ke dalam Bahasa pemrograman dan

pengujian dilakukan (Pressman, 2001). Pengembangan perangkat lunak terbagi menjadi beberapa tahapan yang harus ada, yaitu rekayasa kebutuhan, perancangan, implementasi, dan pengujian. Analisa kebutuhan merupakan proses pengumpulan kebutuhan secara lengkap untuk diidentifikasi kebutuhan yang harus terdapat pada sistem. Perancangan merupakan Proses pemodelan kebutuhan yang sudah di dapatkan. Implementasi merupakan proses untuk menerjemahkan perancangan ke dalam Bahasa pemrograman. Pengujian merupakan proses untuk memastikan perangkat lunak berjalan dengan semestinya dan sesuai dengan apa yang direncanakan.

Metode untuk pengembangan perangkat lunak dapat dikerjakan dengan menggunakan metode konvensional atau terstruktur dan berorientasi objek (Prerssman, 2001). Metode konvensional atau metode terstuktur adalah metode dengan menjelaskan aliran data pada perancangannya. Metode berorientasi object adalah metode dengan mendefinisikan objek-objek yang di butuhkan pada domain masalah. Metode pengembangan yang akan dilakukan pada sistem manajemen bahan baku dan pemesanan pada Sablon Malang akan menggunakan berorientasi objek. Karena dengan menggunakan metode berorientasi objek maka pembangunan perangkat lunak bisa dilakukan dengan cepat dan juga memudahkan dalam melakukan perubahan program. Proses pembangunan perangkat lunak bisa dilakukan dengan cepat karena metode berorientasi objek menggunakan konsep *reusability* yaitu penggunaan kembali komponen program.

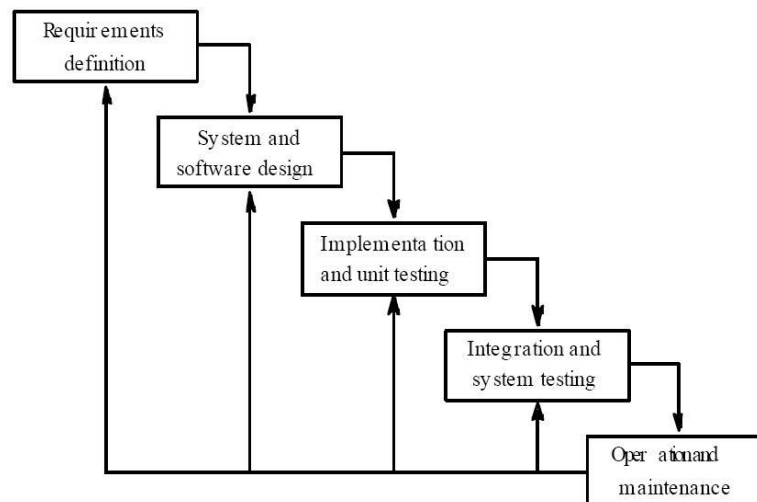
2.3.1 Model Pengembangan perangkat Lunak

Proses pengembangan perangkat lunak dapat menggunakan model-model pengembangan perangkat lunak yang sering disebut dengan *Software Development Life Cycle (SDLC)* (Pressman, 2001). SDLC merupakan sebuah siklus hidup pengembangan perangkat lunak yang terdiri dari beberapa tahapan-tahapan penting dalam membangun perangkat lunak yang dilihat dari segi pengembangannya. Dengan siklus SDLC, proses membangun sistem dibagi menjadi beberapa langkah dan pada sistem yang besar, masing-masing langkah dikerjakan oleh tim yang berbeda. SDLC tidak hanya penting untuk proses produksi software, tetapi juga sangat penting untuk proses *maintenance software* itu sendiri. Terdapat beberapa model pada pengembangan perangkat lunak antara lain *waterfall*, *prototyping*, dan RAD. Model yang akan digunakan dalam pengembangan sistem manajemen bahan baku dan pemesanan pada Sablon Malang adalah *waterfall* karena kebutuhan pada sistem tersebut sudah didapatkan pada tahap awal. Syarat model *waterfall* sendiri adalah semua kebutuhan harus sudah teridentifikasi dengan jelas pada tahapan awal yaitu tahap rekayasa kebutuhan (Pressman, 2001).

2.3.1.1 Waterfall Model

Waterfall model merupakan salah satu model pengembangan perangkat lunak yang sering dipakai. Model ini memiliki tahapan yang dimulai dari rekayasa kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan (Pressman, 2001). Pada model *waterfall* tahapan pengembangan dapat berjalan jika tahapan sebelumnya sudah terselesaikan. Dokumentasi pada *waterfall* model dilakukan pada setiap tahapan pengembangan.

Secara umum tahapan pada model *waterfall* dapat dilihat pada gambar berikut :



Gambar 2.1 Tahapan *waterfall* model

Sumber: Pressman (2001)

1. Rekayasa Kebutuhan

Rekayasa kebutuhan merupakan tahapan untuk mendefinisikan kebutuhan sistem berdasarkan apa yang dibutuhkan oleh user. Tahapan Rekayasa kebutuhan harus didokumentasikan agar user dapat melihat proses analisis dan mengevaluasi jika terdapat kebutuhan yang tidak terpenuhi (Pressman, 2001). Terdapat dua kebutuhan yang harus terpenuhi oleh sistem, yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional adalah suatu fungsi yang harus disediakan oleh sistem, bagaimana sistem harus memberi reaksi terhadap data yang masuk, dan bagaimana sistem harus berjalan dalam beberapa situasi, kebutuhan non-fungsional merupakan batasan yang diberikan oleh sistem seperti *usability*, *security*, *performance*, dan *compatibility* (Sommerville, 2011). Pada analisis kebutuhan terdapat beberapa cara untuk mengumpulkan informasi yang diinginkan. Teknik yang dipakai untuk mengumpulkan informasi yaitu :

a. Observasi

Observasi terdapat dua jenis observasi yang signifikan yaitu *participant observation* (observasi partisipan) dan *direct observation* (observasi langsung). Secara umum *participant observation* adalah peneliti menjadi pemain aktif dalam lingkungan penelitian sedangkan *direct observation* adalah peneliti hanya menjadi pengamat langsung perilaku tersebut tanpa terlibat langsung (Bandur, 2016 : 105).

b. Wawancara

Wawancara banyak digunakan dalam sebuah penelitian. Metode wawancara terdiri dari *unstructured interviews* dan *semi-structured interview*. Dalam *unstructured interviews* peneliti tidak menyiapkan pertanyaan penuntun sebelum melakukan wawancara, tetapi cukup menyediakan tema umum yang hendak didalami dari informan. Sedangkan *semi-structured interview* peneliti

menyiapkan pertanyaan penuntun untuk dijadikan panduan utama dalam wawancara (Bandur, 2016 : 106).

2. Perancangan

Perancangan merupakan tahapan yang menerjemahkan kebutuhan yang didapat ke dalam bentuk pemodelan yang terfokus pada struktur data, arsitektur sistem, antarmuka, dan komponen. Perancangan struktur data mengubah informasi dari hasil rekayasa kebutuhan menjadi struktur data. Perancangan arsitektur adalah mendefinisikan elemen utama dan relasi antar elemen utama tersebut. Perancangan antar muka adalah menjelaskan bagaimana perangkat lunak berkomunikasi dengan manusia sebagai pemakainya. Perancangan komponen yang detail. Tahapan perancangan juga didokumentasikan sama seperti tahap rekayasa kebutuhan.

3. Implementasi

Implementasi merupakan tahapan yang menerjemahkan perancangan kedalam Bahasa pemrograman yang dapat di baca oleh mesin. Implementasi yang di lakukan menggunakan Bahasa pemrograman seperti Java, C++, PHP, dan lain-lain. Tahapan implementasi juga di didokumentasikan seperti tahapan lainnya.

4. Pengujian

Pengujian merupakan tahapan yang dilakukan setelah implementasi. Pengujian perangkat lunak dilakukan untuk mengetahui apakah sistem yang sudah dibuat telah memenuhi kebutuhan yang sudah teridentifikasi dan memastikan bahwa sistem tidak terdapat kesalahan. Pengujian perangkat lunak memiliki stategi yang dapat dilakukan diantaranya adalah :

A. Pengujian Unit

Pengujian unit merupakan pengujian yang fokus pada klas-klas atau modul-modul perangkat lunak.

B. Pengujian Integrasi

Pengujian Integrasi merupakan pengujian yang fokus pada hubungan antar klas dan bertujuan untuk memastikan bahwa tidak terdapat kesalahan pada saat menggabungkan klas yang berhubungan.

C. Pengujian Validasi

Pengujian validasi merupakan pengujian yang dilakukan untuk memastikan bahwa semua kebutuhan pada sistem telah terpenuhi dan tidak terdapat kesalahan pada sistem.

Pengujian perangkat lunak terdapat dua teknik pengujian yang dapat dilakukan, yaitu pengujian yang dilakukan untuk pengujian unit atau bisa disebut *white box testing* dan pengujian yang dilakukan untuk pengujian validasi yaitu *black box testing*.

A. White Box Testing

White box testing merupakan jenis pengujian perangkat lunak yang lebih berfokus pada struktur program dan memastikan bahwa semua jalur independen telah dieksekusi setidaknya sekali. Terdapat beberapa jenis *white box testing*, diantaranya (Pressman, 2001).

a. Basis Path Testing

Pengujian untuk mengetahui jumlah jalur eksekusi yang harus dilewati berdasarkan pengukuran *cyclomatic complexity*. Sebelum menghitung

cyclomatic complexity, terlebih dahulu menggambar alur algoritmanya. Suatu kondisi pada algoritma digambarkan oleh node dan panah atau *edge* akan digambarkan oleh garis lurus. *Cyclomatic complexity* didapatkan dari *edge* dikurangi *node* lalu ditambah dengan dua. Bisa juga didapatkan dari jumlah *node* bercabang ditambah satu. *Test case* akan dibuat pada setiap jaluk eksekusi.

b. *Condition Testing*

Pengujian *condition testing* adalah pengujian yang mendesain *test case* berdasarkan kondisi logis yang ada pada suatu program. Kondisi sederhana adalah variable Boolean atau suatu persamaan relasional. Operator relasional adalah salah satu operator dari "<",">","=","≠","<=",">=". Kondisi tanpa persamaan relasional merupakan persamaan *boolean*. *Condition testing* merupakan pengujian yang berfokus pada kondisi program.

c. *Loop Testing*

Pengujian ini berfokus pada pernyataan perulangan pada konstruksi program. Terdapat empat jenis perulangan yaitu *simple loops*, *nested loops*, *concatenated loops*, dan *unstructured loops*. Pengujian *simple loops* dapat dilakukan dengan meloncati seluruh *loop*, hanya satu yang melalui *loop*, dua yang melalui *loop*, m melewati *loop* dimana m kurang dari n, dan n-1, n, n+1 melewati *loop*. Pengujian *nested loops* dapat dimulai dari lingkaran terdalam dan melakukan *simple loops testing* untuk *loop* terdalam. Pengujian *concatenated loops* dapat dilakukan menggunakan *simple loops testing* untuk *loop* independen dan dapat menggunakan *nested loops testing* untuk *loop* yang tidak independen. Jika terdapat *unstructured loops*, sebaiknya *loop* dirancang ulang.

B. *Black Box Testing*

Black box testing merupakan pengujian perangkat lunak yang tidak memperhatikan struktur program. Pengujian ini hanya memperhatikan spesifikasi kebutuhan dari perangkat lunak yang telah didefinisikan pada tahapan rekayasa kebutuhan, memeriksa apakah *input* dan *output* sesuai dengan yang diinginkan user atau tidak dan memeriksa kesalahan pada antarmuka. Terdapat beberapa jenis *black box testing*, diantaranya (Pressman, 2001) :

a. *Graph-Based Testing*

Pengujian ini memeriksa objek-objek dan hubungan antar objek tersebut. Objek akan digambarkan dalam bentuk *node* dan hubungannya akan digambarkan oleh garis lurus atau *link weight*. Setiap objek harus berhubungan dengan objek lainnya. Pengujian dilakukan untuk memastikan bahwa tidak terdapat *error* pada hubungan antar objek.

b. *Scenario-Based Testing*

Pengujian ini memusatkan pada pada yang dilakukan oleh pengguna. Pengujian ini dilakukan untuk mengungkapkan kesalahan pada saat pengguna berinteraksi dengan sistem. Pengujian ini menerapkan *use case scenario* untuk menentukan apa yang harus pengguna lakukan sebagai acuan.

c. *Equivalence Partitioning*

Pengujian ini membagi domain masukan ke dalam kelas-kelas data, lalu memeriksa kembali hasil yang dikeluarkan sistem berdasarkan masukan tersebut apakah sesuai atau tidak dengan yang diharapkan.

d. *Boundary Value Analysis*

Pengujian ini berguna untuk melakukan pengujian terhadap *boundary values* atau nilai tiap sisi dari batasan. *Boundary values* diusahakan mempunyai selisih sekecil mungkin dari nilai batasan.

e. *Bottom Up Testing*

Pengujian ini adalah pengujian terpadu dimana komponen tingkat terendah diuji terlebih dahulu, kemudian digunakan untuk memfasilitasi pengujian komponen tingkat yang lebih tinggi. Proses ini diulang sampai komponen di bagian atas hirarki diuji.

f. *Comparison Testing*

Pengujian ini adalah pengujian seluruh versi perangkat lunak yang dilakukan secara *parallel* untuk membandingkan hasil keluaran apakah konsisten atau tidak.

Dari berbagai teknik pengujian pada *white box testing* dan *black box testing*, tidak semua teknik pengujian dipakai.

5. Pemeliharaan

Tahapan pemeliharaan merupakan tahapan yang dilakukan jika dibutuhkan perubahan pada sistem. Perubahan dapat terjadi karena terdapat kesalahan pada sistem, diperlukan operasi baru pada sistem atau karena user membutuhkan kebutuhan fungsional baru untuk meningkatkan kinerja pada sistem (Pressman, 2001).

2.3.2 Pendekatan Object Oriented

Pendekatan berorientasi objek atau bisa juga disebut dengan *Object Oriented* (OO) pertama kali dikembangkan sekitar pada tahun 1960 (Pressman, 2001). OO digunakan pada pembangunan perangkat lunak menggunakan Bahasa pemrograman yang berorientasi objek seperti *Java*, *C++*, *Eiffel*, dan *Smalltalk*. Kelebihan dari *Object Oriented* adalah mudah dalam melakukan perubahan program, konsisten antara analisis dan perancangan, dan waktu pengembangan perangkat lunak yang termasuk cepat. Proses pembangunan perangkat lunak menggunakan OO dimulai dari rekayasa kebutuhan untuk mengidentifikasi kebutuhan perangkat lunak yang bisa diambil dari kebutuhan user, objek-objek, dan klas-klas dari domain masalah. Kemudian proses selanjutnya adalah perancangan berdasarkan apa yang sudah didapatkan pada rekayasa kebutuhan. Perancangan yang dilakukan dalam OO adalah perancangan arsitektur, antarmuka, dan komponen. Proses selanjutnya yaitu mengubah rancangan menjadi kode program yang dapat dibaca oleh mesin yang menggunakan Bahasa pemrograman berorientasi objek. Proses terakhir yaitu pengujian yang disini menguji perancangan arsitektur, antarmuka, dan komponen (Pressman, 2001).

2.3.2.1 Konsep Object Oriented

Konsep pendekatan berorientasi objek atau OO membagi domain masalah ke dalam bentuk objek nyata dimana masing-masing objek memiliki atribut yang mendefinisikan object tersebut dan perilaku yang spesifik (Pressman, 2001). Pada pendekatan OO object

yang sudah teridentifikasi nantinya akan menjadi klas-klas atau sub klas. Setiap klas akan memiliki sekumpulan atribut dan sekumpulan operasi yang mendeskripsikan perilaku dari klas tersebut. Terdapat tiga konsep pada pendekatan OO, yaitu *enkapsulasi*, *inheritance*, dan *polymorphism*. *Enkapsulasi* adalah membungkus atribut dan operasi pada suatu klas agar tidak dapat diakses secara sembarangan. *Inheritance* adalah suatu hubungan dimana anak klas dapat mewarisi atribut dan operasi dari induk klas. *Polimorphysm* adalah setiap anak klas memungkinkan memiliki nama operasi yang sama dengan operasi yang ada pada induk klas (Pressman, 2001).

Proses pertama yang dilakukan pada pengembangan perangkat lunak menggunakan pendekatan berorientasi objek yaitu dengan *object oriented analysis* (OOA). OOA mengidentifikasi kebutuhan, objek-objek, atribut, dan perilaku yang mendeskripsikan objek tersebut berdasarkan domain masalah yang telah dideskripsikan oleh user. OOA menggunakan *UML* pada permodelanya, yaitu *use case* yang digunakan untuk memodelkan kebutuhan pada OOA. Lalu *use case scenario* yang digunakan untuk menjelaskan bagaimana sistem digunakan. Proses selanjutnya setelah OOA adalah *object oriented design* (OOD). OOD mentransformasikan model analisis yang telah dibuat pada OOA kedalam bentuk model perancangan. OOD juga menggunakan *UML* untuk memodelkan perancangannya. Perancangan yang dilakukan pada OOD adalah perancangan sistem, perancangan objek, dan perancangan antarmuka. Perancangan sistem menggambarkan struktur dari klas dan relasi antar klas tersebut menggunakan *class diagram*. Perancangan objek yang mendeskripsikan objek-objek dan interaksinya satu sama lain dengan menggunakan *state diagram*, *sequence diagram*, dan *activity diagram*. Proses selanjutnya setelah OOD adalah *object oriented programing* (OOP). OOP mengubah model perancangan kedalam kode yang dapat dijalankan oleh mesin berupa bahasa pemrograman yang mengimplementasikan OOP. Proses selanjutnya yang terakhir adalah *object oriented testing* (OOT). OOT adalah pengujian yang dilakukan setelah perancangan selesai. Pengujian yang dilakukan oleh OOT adalah pengujian unit, pengujian integrase, dan pengujian validasi (Pressman, 2001).

2.3.2.2 Permodelan Object Oriented





Permodelan OOA dan OOD dimodelkan menggunakan diagram yang terdapat pada *UML* (Pressman, 2001). *UML* yang disebut juga Bahasa permodelan merupakan sebuah notasi yang digunakan untuk menggambarkan suatu perancangan perangkat lunak pada pengembangan perangkat lunak yang menggunakan OO. Terdapat tiga diagram *UML* yang digunakan dalam pengembangan sistem manajemen bahan baku dan pemesanan (Flowler, 2005):

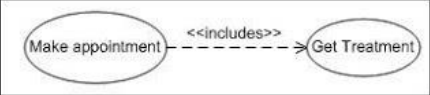
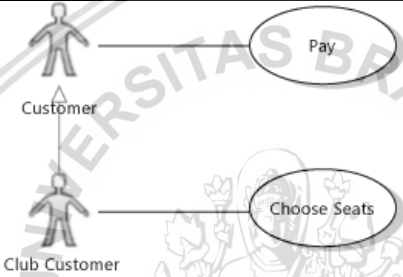
1. *Use Case Diagram*

Use case Diagram digunakan untuk menggambarkan interaksi antara sistem, sistem eksternal, dan pengguna. Dengan kata lain, secara grafis menjelaskan apa yang dilakukan sistem serta secara grafis mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna mengharapkan interaksi dengan sistem ini (Lethbridge, et al., 2004).

Penjelasan mengenai komponen-komponen dari *Use Case Diagram* dapat dilihat pada Tabel 3.1.

Tabel 2.1 Komponen Use Case Diagram

No	Komponen Komponen <i>Use Case</i>	Gambar	Pengertian
1	<i>Use Case</i>		<i>Use case</i> merupakan urutan langkah-langkah yang secara tindakan saling terkait (skenario), baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal.
2	Aktor (Pelaku)		Pelaku adalah segala sesuatu yang perlu berinteraksi dengan sistem untuk pertukaran informasi. Pelaku menginisiasi kegiatan sistem, yakni sebuah <i>use case</i> .
4	<i>Association</i>		<i>Association</i> merupakan hubungan antara seorang pelaku dengan satu <i>use case</i> , terbentuk kapan pun <i>use case</i> menggambarkan interaksi antara keduanya.
5	<i>Extend</i>		<i>Extends</i> merupakan relasi <i>use case</i> tambahan ke sebuah <i>use case</i> yang mana <i>use case</i> yang ditambahkan dapat

			berdiri sendiri walaupun tanpa <i>use case</i> tambahan tersebut.
6	<i>Include</i>		<i>Include</i> merupakan kelakuan yang harus terpenuhi agar sebuah <i>event</i> dapat terjadi, yang mana pada kondisi ini sebuah <i>use case</i> adalah bagian dari <i>use case</i> lainnya.
7	<i>Inheritance</i>		<i>Inheritance</i> merupakan hubungan dua atau lebih pelaku berbagi kelakuan umum. Hal ini dilakukan untuk mengurangi komunikasi redundan dengan sistem.

Sumber: Fowler (2005)

2. Sequence Diagram

Sequence diagram adalah diagram yang menggambarkan alur control diantara objek-objek yang terdapat pada *use case*. Alur control tersebut mendeskripsikan operasi apa saja yang berjalan (Page-Jones, 2002).

Fungsi serta peran *sequence diagram* diantaranya :


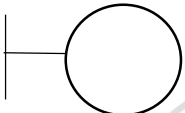


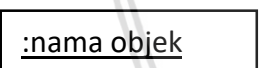
1. *Overview* perilaku sistem.
2. Menunjukkan objek – objek yang diperlukan.
3. Mendokumentasikan skenario dari suatu diagram *use case*.
4. Memeriksa jalur – jalur pengaksesan.

Tujuan *sequence diagram* adalah sebagai berikut :

1. Merupakan diagram yang paling cocok untuk mengembangkan model Deskripsi *use-case* menjadi spesifikasi design.
2. Analisa dan desain, memfokuskan pada identifikasi method didalam sebuah system.

Dengan memahami simbol-simbol ini diharapkan dapat mempermudah kita dalam penggambaran diagram sesuai dengan *use case* yang kita buat. Untuk tanda yang digunakan dalam diagram ini adalah sebagai berikut :

Tabel 2.2 Sequence Diagram

No	Nama simbol	Definisi
1		Merupakan representasi aktor dalam aplikasi.
2		Merupakan <i>boundary class</i> yang menghubungkan antara aktor dengan kelas <i>stereotype</i> ataupun objek lainnya.
3		Merupakan <i>entity class</i> dan merupakan representasi dari sebuah entitas.
4		Merupakan <i>control class</i> dan merupakan representasi kontrol.
5		Merupakan representasi objek.

Sumber: Fowler (2005)

3. Class Diagram

Class diagram adalah sebuah *diagram* yang menggambarkan struktur dan penjelasan class, obyek serta hubungan satu sama lain. *Class diagram* juga menjelaskan hubungan antar class dalam sebuah sistem yang dibuat dan bagaimana cara agar terkolaborasi untuk mencapai suatu tujuan (Page-Jones, 2002).

Class diagram juga memiliki 3 area pokok (utama) yaitu : nama, atribut, dan operasi. Nama berfungsi untuk member identitas pada sebuah kelas, atribut fungsinya adalah untuk member karakteristik pada data yang dimiliki suatu objek di dalam kelas, sedangkan operasi fungsinya adalah memberikan sebuah fungsi ke sebuah obyek. Dalam mendefinisikan metode yang ada di dalam kelas harus diperhatikan yang namanya Cohesion dan Coupling. Cohesion adalah ukuran keterkaitan sebuah instruksi di sebuah metode, Coupling adalah ukuran keterkaitan antar metode.

Di dalam *class diagram* terdapat hubungan antar kelas secara konseptual, yang disebut Relasi antar *Class*, di UML disediakan macam-macam relasi antar *Class*, di

antaranya: *Asosiasi* (Hubungan statis antar kelas), *Agregasi* (hubungan dari keseluruhan objek), *Generalisasi* (relasi beberapa subkelas ke *super* kelas), dan *Dependency* (keterhubungan tiap kelas) (Ambler, 2012).

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut :




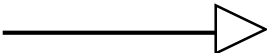
- Kelas Main. Yaitu kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan. Kelas Interface.
- Kelas yang mendefinisikan dan mengatur tampilan ke pemakai. Biasanya juga disebut kelas boundaries.
- Kelas yang diambil dari pendefinisian *use case*. Merupakan kelas yang menangani fungsi-fungsi yang harus ada dan diambil dari pendefinisian *use case*.
- Kelas Entitas. Merupakan kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Atribut dan metode dapat memiliki salah satu sifat berikut :

- Private*, tidak dapat dipanggil dari luar class yang bersangkutan.
- Protected*, hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya.
- Public*, dapat dipanggil oleh siapa saja.

Dalam class diagram memiliki beberapa hubungan antar class, untuk relasinya bisa dilihat pada tabel berikut :

Tabel 2.3 Relasi Class Diagram

No	Nama simbol	Definisi
1	<div> <div>Kelas</div> <div> <div>nama_kelas</div> <div>+atribut()</div> <div>+operasi()</div> </div> </div>	Merupakan kelas pada struktur sistem.
2	<div> <div>Interface</div> <div>  </div> </div>	Merupakan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3	<div> <div>Asosiasi</div> <div>  </div> </div>	Merupakan relasi antar kelas.
4	<div> <div>Asosiasi berarah</div> <div>  </div> </div>	Merupakan relasi antar kelas yang bermakna kelas yang satu digunakan oleh kelas yang lain.
5	<div> <div>Generalisasi</div> <div>  </div> </div>	Merupakan Relasi antar kelas dengan prinsip generalisasi.

6	Agregasi →	Merupakan relasi antar kelas bermakna semua bagian.
7	Kebergantungan →	Merupakan relasi kelas yang bergantung dengan kelas lain.

Sumber: Fowler (2005)

2.4 Teknologi Pengembangan Sistem

Terdapat beberapa teknologi yang digunakan untuk membangun sistem manajemen bahan baku dan perancangan pada sablon malang, yaitu *Material Requirement Planning*, *PHP Hypertext Preprocessor*, *MySQL*, dan *CodeIgniter*.

2.4.1 Material Requirement Planning

Bahan baku merupakan salah satu faktor penting dalam membuat suatu produk. Untuk menjamin kelancaran produksi, ketepatan waktu penerimaan bahan baku dan bahan pendukung lainnya oleh pihak produksi diperlukan suatu perencanaan. Tanpa perencanaan yang matang serta pengendalian yang ketat, resiko ketepatan waktu dalam Gudangan dan penerimaan material (bahan baku dan bahan pendukungnya) akan menjadi semakin tinggi yang mengakibatkan produksi tidak mampu untuk menghasilkan jumlah unit produk yang dibutuhkan oleh Pelanggan/konsumen (Herjanto, 2008). Oleh karena itu, diperlukan suatu teknik ataupun sistem yang berfungsi untuk merencanakan jadwal keperluan material yang dibutuhkan. Teknik ataupun sistem tersebut biasanya disebut *Material Requirement Plan* atau disingkat dengan MRP.

Menurut Stevenson (2005), Material Requirement Planning (MRP) adalah suatu sistem informasi berbasis komputer yang menterjemahkan Jadwal Produksi Induk (*Master Production Schedule*) untuk barang Jadi (produk akhir) menjadi beberapa tahapan kebutuhan sub-assy, komponen dan bahan baku. Dengan demikian dapat kita katakan bahwa MRP adalah suatu rencana produksi untuk sejumlah produk jadi dengan menggunakan tenggang waktu sehingga dapat ditentukan kapan dan berapa banyak dipesan untuk masing-masing komponen suatu produk yang akan dibuat.

Menurut Heizer dan Render (2014) menyimpulkan untuk menjalankan sistem MRP, ada tiga elemen utama yang harus dimasukkan, yaitu:

1. Master Production Schedule (MPS)

Master Production Schedule atau Jadwal Produksi Induk adalah suatu perencanaan yang terdiri dari tahapan waktu dan jumlah produk jadi yang akan diproduksi oleh sebuah perusahaan manufakturing. MPS ini pada umumnya berdasarkan *order* (pesanan) pelanggan dan perkiraan *order* (Forecast) yang dibuat oleh perusahaan sebelum dimulainya sistem MRP.

Secara garis besar pembuatan suatu MPS biasanya dilakukan atas tahapan-tahapan sebagai berikut :

- a. Identifikasi sumber permintaan dan jumlahnya, sehingga dapat diketahui besarnya permintaan produk akhir setiap periodanya.
- b. Menentukan besarnya kapasitas produksi dan kecepatan operasi yang diperlukan untuk memenuhi permintaan yang telah diidentifikasi, perencanaan ini biasanya dilakukan pada tingkat agregat, sehingga masih merupakan perencanaan global.
- c. Menyusun rencana rinci dari setiap produk akhir yang akan dibuat. Tahap ini merupakan penjabaran dari rencana agregat (global) sehingga akan didapat rencana produksi setiap produk akhir yang dibuat dan perioda waktu pembuatannya.

2. *Bill of Material (BOM)*

BOM adalah sebuah daftar yang berisikan jumlah masing-masing bahan baku, bahan pendukung dan sub-assy (semi produk) yang dibutuhkan untuk membuat suatu produk jadi (Heizer dan Render, 2014). BOM dibuat berdasarkan apa yang terjadi dilapangan dari awal sampai akhir menjadi produk.

3. Ketersediaan Persediaan (on hand)

Data ini menjadi landasan untuk pembuatan MRP karena memberikan informasi tentang jumlah persediaan bahan baku dan barang jadi yang aman (minimum) serta keterangan lainnya, seperti : (1) kapan kita mendapat kiriman barang; (2) berapa jangka waktu pengiriman barang (*lead time*); (3) berapa besar kelipatan jumlah pemesanan barang (*lot size*). Semua keterangan itu mendukung penyusunan MRP yang tepat sehingga sesuai dengan tujuan awalnya untuk merencanakan jumlah dan waktu pesanan bahan baku yang tepat agar proses produksi tidak terlambat. Heizer dan Render (2014) menyebutkan bahwa agar sebuah MRP dapat bekerja dengan baik dibutuhkan suatu manajemen persediaan yang baik. Jika perusahaan belum mencapai setidaknya 99 persen ketelitian catatan, maka perencanaan kebutuhan material tidak akan bekerja dengan baik.

Menggambarkan keadaan dari setiap komponen atau material yang ada dalam persediaan, yang berkaitan dengan :

- a. Jumlah persediaan yang dimiliki pada setiap periode (*on hand inventory*).
- b. Jumlah barang yang sedang dipesan dan kapan pesanan tersebut akan datang (*on order Inventory*).
- c. Waktu ancap – ancap (*lead time*) dari setiap bahan.

Status persediaan ini harus diketahui untuk setiap bahan atau item dan diperbaharui setiap terjadi perubahan untuk menghindari adanya kekeliruan dalam perencanaan.

Keluaran MRP adalah informasi yang dapat digunakan untuk melakukan pengendalian produksi. Keluaran pertama berupa rencana pemesanan yang disusun berdasarkan waktu ancap dari setiap komponen / item. Dengan adanya rencana pemesanan, maka kebutuhan bahan pada tingkat yang lebih rendah dapat diketahui. Selain itu proyeksi kebutuhan kapasitas juga akan diketahui, yang selanjutnya akan memberikan revisi atas perencanaan kapasitas yang dilakukan pada tahap sebelumnya.

Keluaran MRP lainnya adalah :

1. Memberikan catatan pesanan penjadwalan yang harus dilakukan / direncanakan baik dari pabrik maupun dari Gudang.
2. Memberikan indikasi penjadwalan ulang.
3. Memberikan indikasi pembatalan pesanan.
4. Memberikan indikasi keadaan persediaan.

Dengan demikian, pada garis besarnya, MRP bukan hanya menyangkut manajemen material dan persediaan saja, tetap juga mempengaruhi aktivitas perencanaan dan pengendalian produksi sehari-hari di perusahaan.

MRP pada dasarnya bertujuan untuk merancang suatu sistem yang mampu menghasilkan informasi untuk mendukung tindakan yang tepat baik berupa pembatalan pesanan, pesan ulang, atau penjadwalan ulang. Tindakan ini sekaligus merupakan suatu pegangan untuk melakukan pembelian bahan baku atau produksi. Menurut Herjanto (2007:276) dalam bukunya yang berjudul "Manajemen Operasi", mengemukakan beberapa tujuan dari MRP, diantaranya:

1. Meminimalkan persediaan, MRP menentukan berapa banyak dan kapan suatu komponen diperlukan disesuaikan dengan jadwal induk produksi (*Master Production Schedule*). Dengan menggunakan metode ini, pengadaan (pembelian) atas komponen-komponen yang diperlukan untuk suatu rencana dapat dilakukan sebatas yang diperlukan saja sehingga dapat meminimalkan biaya persediaan.
2. Mengurangi resiko karena keterlambatan produksi atau pengiriman. MRP mengidentifikasi banyaknya bahan dan komponen yang diperlukan baik dari segi jumlah dan waktunya dengan memperhatikan waktu tenggan produksi maupun pengadaan komponen, sehingga dapat memperkecil resiko tidak tersedianya bahan yang akan diproses yang dapat mengakibatkan terganggunya rencana produksi.
3. Dengan MRP, jadwal produksi diharapkan dapat dipenuhi sesuai dengan rencana, sehingga komitmen terhadap pengiriman barang dapat dilakukan secara realistis. Hal ini mendorong meningkatnya kepuasan dan kepercayaan konsumen.
4. Meningkatkan efisien, MRP juga mendorong peningkatan efisien karena jumlah persediaan, waktu produksi, dan waktu pengiriman barang dapat direncanakan lebih baik sesuai dengan jadwal induk produksi.

2.4.2 PHP Hypertext Preprocessor (PHP)

PHP merupakan salah satu bahasa pemrograman yang banyak digunakan dan terkenal. Dengan menggunakan PHP *website* menjadi lebih dinamis karena PHP dapat berbasis *database*. Salah satu keunggulan PHP adalah PHP terkenal aman dibandingkan Bahasa pemrograman lainnya dan juga PHP bisa didapat secara gratis (Wardana, 2010). PHP juga salah satu Bahasa pemrograman yang mendukung OOP (*Object Oriented Programming*) sehingga dapat lebih mudah digunakan dibandingkan dengan *Procedural*. PHP juga termasuk Bahasa yang menyatu antara *script* dan *HTML (Hypertext Markup Language)* dan dijalankan pada sisi *server*. *Sintaks PHP* yang dibuat akan sepenuhnya dijalankan pada *server* dan hasilnya akan dikirim ke *Browser*.

2.4.3 MySQL

MySQL dikembangkan oleh perusahaan swedia bernama MySQL AB yang pada saat ini bernama Tcx DataKonsult AB sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak tahun 1979. Awalnya Tcx merupakan perusahaan pengembang software dan konsultan database, dan saat ini MySQL sudah diambil alih oleh Oracle Corp (Arief, 2011).

Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja query cepat, dan mencukupi untuk kebutuhan database perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga bersifat open source (tidak berbayar).

MySQL merupakan database yang pertama kali didukung oleh bahasa pemrograman script untuk internet (PHP dan Perl). MySQL dan PHP dianggap sebagai pasangan software pembangun aplikasi web yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman script PHP.

2.4.4 CodeIgniter

Menurut Sidik (2012), CodeIgniter adalah aplikasi open source yang berupa framework dengan model MVC (*Model View Controller*) untuk membangun *website* dinamis dengan menggunakan PHP. *CodeIgniter* memudahkan *developer* untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal. *CodeIgniter* pertama kali dirilis pada 28 Februari 2006.

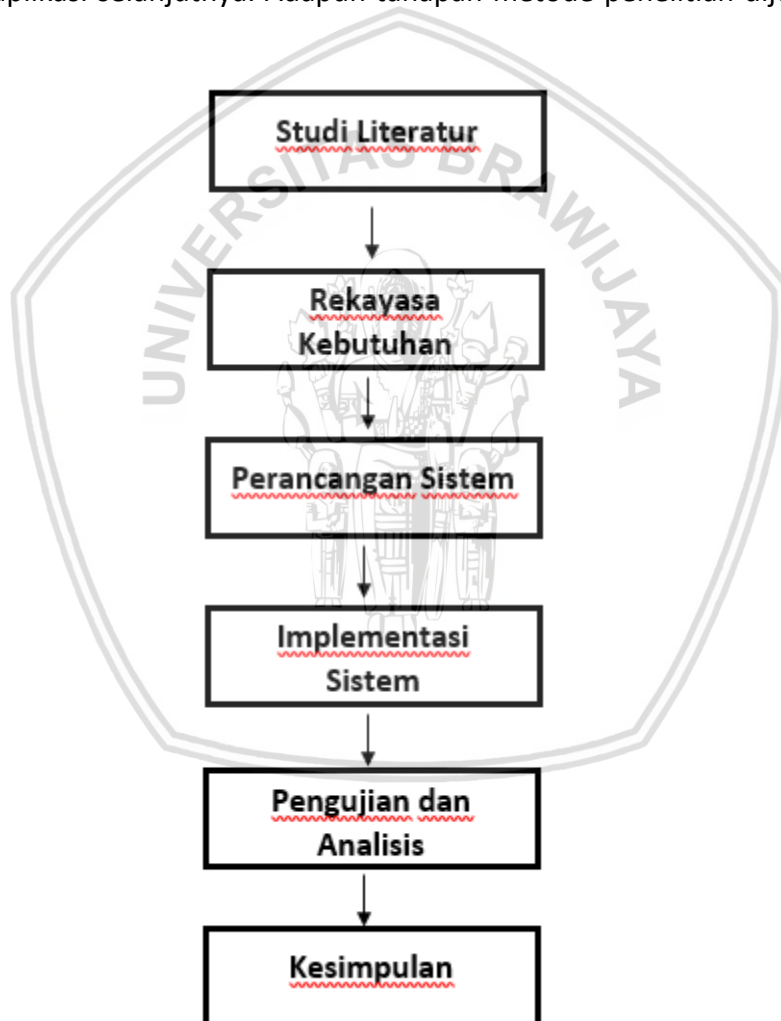
Menurut Blanco & Upton (2009:7) *CodeIgniter* adalah powerful open source PHP framework yang mudah dikuasai, dibangun untuk PHP *programmers* yang membutuhkan toolkit sederhana dan baik untuk membuat *full-featured web applications*. *CodeIgniter* adalah MVC *framework* yang di design untuk mempermudah penggunaanya. Sistem ini akan menggunakan *CodeIgniter* karena bahasa pemrograman yang digunakan adalah PHP jadi akan mudah untuk membangun sebuah sistem.

Selain itu *CodeIgniter* juga memiliki fitur-fitur lainnya yang sangat bermanfaat, antara lain (Daqiqil, 2011)

1. Menggunakan pola MVC
Dengan menggunakan pola MVC ini, struktur kode yang dihasilkan menjadi lebih terstruktur dan memiliki standart yang jelas.
2. *URL Friendly*
URL yang dihasilkan sangat mudah. *CodeIgniter* meminimalisasi pengguna \$GET dan digantikan URL.
3. Kemudahan
Kemudahan dalam mempelajari, membuat *library* dan *helper*, memodifikasi dan mengintegrasikan *library* dan *helper*.

BAB 3 METODOLOGI

Pada bab metodologi ini menjelaskan langkah – langkah yang dilakukan dalam perancangan, implementasi dan pengujian dari sistem “Pengembangan Aplikasi Manajemen Bahan Baku dan Perancangan Pada Sablon Malang” yang akan dibuat. Hal pertama yang harus dilakukan adalah mencari dan memilih studi literature yang tepat, dilanjutkan dengan merekayasa kebutuhan yang diperlukan. Setelah itu melakukan perancangan sistem yang sesuai dengan kebutuhan yang telah disetujui yang nantinya kana menjadi acuan dalam proses implementasi. Kemudian dilakukan pengujian dan analisis sistem yang akan dilakukan untuk mendapatkan kesimpulan dan saran dicantumkan sebagai catatan maupun evaluasi atas sistem yang memungkinkan arah pengembangan aplikasi selanjutnya. Adapun tahapan metode penelitian dijelaskan pada Gambar 3.1.



Gambar 3.1 Tahapan Metode Penelitian

3.1 Studi Literatur

Studi literatur adalah metode untuk mendalami konsep yang digunakan dalam membangun sebuah sistem. Studi literatur berisi landasan teori yang terkait dengan konsep tersebut. Sangat perlu dilakukan agar pengetahuan dasar untuk membangun sebuah sistem terpenuhi dengan baik. Studi literatur yang digunakan dalam membangun sistem aplikasi manajemen bahan baku dan perancangan ini di dapat dari buku, skripsi, artikel ilmiah, dan informasi yang tersedia di internet baik berupa proyek maupun artikel. Studi literatur yang digunakan meliputi di bawah ini :

1. Penelitian Terdahulu
2. Pengembangan Perangkat Lunak
 - a. *Waterfall Model*
 - b. Pendekatan *Object Oriented*
3. Teknologi Pengembangan Sistem
 - a. *Material Requirment Planning*
 - b. PHP, HTML
 - c. *CI Framework*
 - d. DBMS MySQL
4. UML (*Unified Modeling Language*)
 - a. *Use case Diagram*
 - b. *Sequence Diagram*
 - c. *Class Diagram*
5. Proses pengujian sistem
 - a. *Black Box Testing*
 - b. *White Box Testing*

3.2 Rekayasa Kebutuhan

Rekayasa kebutuhan adalah tahapan pertama dari *waterfall model*, tahap ini dilakukan untuk mengetahui kebutuhan apa saja yang dibutuhkan untuk membangun sistem, mulai dari kebutuhan perangkat baik dari segi hardware, software, kebutuhan data, kebutuhan fungsional, maupun kebutuhan non-fungsional. Proses rekayasa akan dilakukan dengan pendekatan OOA (*Object Oriented Analysis*). Kebutuhan yang didapat akan dibuat pemodelan dengan menggunakan salah satu diagram UML yaitu *use case diagram*. Pemodelan kebutuhan dilakukan untuk menjelaskan kebutuhan sistem yang akan dibangun dan siapa saja yang berinteraksi dengan sistem tersebut. Setelah itu dilakukan pembuatan *use case scenario* untuk menjelaskan cara kerja sistem berdasarkan *use case diagram* yang telah dibuat. Proses rekayasa kebutuhan aplikasi manajemen bahan baku dan pemesanan sablon malang adalah sebagai berikut :

1. Proses Elisitasi Kebutuhan
 - a. Observasi

Pengumpulan informasi yang penulis lakukan salah satunya adalah observasi. Penulis melakukan observasi dikarenakan untuk mengetahui secara langsung keadaan dari sablon malang dan mengetahui gambaran secara realistis. Pada tahapan pengumpulan data observasi menggunakan *direct observation* terhadap

permasalahan yang dihadapi guna melengkapi data-data yang diperlukan dalam penelitian.

b. Wawancara

Selain menggunakan observasi pengumpulan informasi dilakukan dengan menggunakan metode wawancara. Metode ini dirasa tepat karena pemilik dari sablon malang sudah mengerti tentang sebuah sistem. Metode wawancara dilakukan peneliti adalah *unstructured interviews* yang memiliki tema tentang proses pemesanan beserta manajemen bahan baku yang terjadi di sablon malang. peneliti mengadakan tanya jawab kepada narasumber yang bersangkutan yaitu pemilik sablon malang dan kasir.

2. Proses Spesifikasi Kebutuhan

Proses ini mendefinisikan kebutuhan hingga menjadi kebutuhan yang spesifik. Dalam proses ini juga melakukan identifikasi aktor yang akan menggunakan sistem. Kebutuhan yang didapat akan dibagi menjadi kebutuhan fungsional dan non-fungsional untuk mengetahui lebih rinci aktifitas apa saja yang bisa dilakukan oleh pengguna maupun sistem dan batasan sistem yang akan dibangun. Pada proses ini juga melakukan pemodelan kebutuhan yang berupa *use case diagram* dan *scenario diagram*.

3. Proses Validasi Kebutuhan

Proses validasi dilakukan dengan pengecekan terhadap kebutuhan yang telah didefinisikan dan dispesifikasikan dengan mencocokkan kembali hasil dari observasi dan wawancara yang dilakukan di elisitasi kebutuhan.

4. Proses Manajemen Kebutuhan

Proses manajemen kebutuhan dilakukan control terhadap kebutuhan yang sudah didefinisikan dan memberikan kode pada kebutuhan dalam menulis kebutuhan sistem dengan kode MKB_XX_YYYY. MKB menunjukan nama sistem sedangkan XX menunjukan nomor kebutuhan fungsional atau non fungsional dan YYYY menupakan nomor kebutuhan.

3.3 Perancangan Sistem

Perancangan sistem adalah tahapan *waterfall model* setelah dilakukanya rekayasa kebutuhan, tahap ini dilakukan setelah semua kebutuhan sudah terpenuhi pada rekayasa kebutuhan sistem. Perancangan sistem menggunakan metode OOD (*Object Oriented Design*) dan menggunakan UML (*Unified Modeling language*). Proses ini bertujuan untuk menerjemahkan kebutuhan yang didapat dari rekayasa kebutuhan yang didapat kedalam bentuk permodelan yang terfokus pada arsitektur sistem, struktur data, antar muka dan komponen. Proses perancangan sistem yang akan dibangun yaitu :

1. Perancangan Arsitektur

Perancangan arsitektur akan menggunakan UML (*Unified Modeling language*) yaitu *sequence diagram* dan *class diagram*. Pada *sequence diagram* akan diambil 3 *use case* yang nantinya dibuat menjadi *sequence diagram* yang mana merupakan fitur utama dari sistem.

2. Peancangan Struktur Data

Perancangan struktur data terdapat rancangan tabel basis data yang akan digunakan oleh sistem dalam bentuk ERD (*Entity Relationship Diagram*) dan PDM (*Physical Data Model*) yang nantinya akan dijadikan dasar dalam pembuatan database sistem.

3. Perancangan Antar Muka

Perancangan antar muka terdapat rancangan antar muka dari sistem dalam bentuk tata muka komponen yang ada pada sistem, nantinya akan digunakan dasar dalam implementasi antar muka sistem.

4. Perancangan Komponen

Perancangan komponen terdapat 3 algoritma utama dalam sistem yang akan dibangun. Algoritma tersebut akan ditulis dalam bentuk *pseudocode*.

3.4 Implementasi

Implementasi merupakan tahap yang mana kode program akan dibuat berdasarkan desain sistem yang sebelumnya telah dirancang. Tahapan ini dilakukan setelah menyelesaikan tahap perancangan sistem yang termasuk urutan dari *waterfall model*. Sistem dilakukan dengan menerapkan metode *OOP (Object Oriented Programming)*. Tahapan-tahapan yang dilakukan dalam implementasi sistem di antaranya sebagai berikut:

1. Implementasi Bahasa pemrograman menggunakan PHP dengan menggunakan framework *CodeIgniter*.
2. Implementasi basis data, dengan menggunakan DBMS MySQL pada server localhost untuk mempermudah manipulasi dan penyimpanan dataset.
3. Implementasi akses sistem, dengan menggunakan browser Mozilla Firefox, Google Chrome.
4. Implementasi antar muka akan menggunakan Bahasa HTML, CSS, Bootstrap, dan Javascript.

3.5 Pengujian dan Analisis

Pengujian dan Analisis adalah tahapan terakhir dari *waterfall model*. Tahap pengujian ini merupakan tahap yang bertujuan untuk menguji apakah sistem sudah sesuai dengan kebutuhan yang sudah teridentifikasi dan sistem tidak terdapat kesalahan meliputi *bug*, *error* atau *defect*. Pengujian tersebut meliputi :

1. Pengujian Unit digunakan untuk menguji setiap unit terkecil dari sistem, yang mana unit bisa berupa komponen, *class*, atau object dari sistem berdasarkan perancangan komponen yang ada. Pengujian unit dilakukan dengan menggunakan *basis path testing* yang merupakan salah satu jenis pengujian dari *white box testing* Pengujian unit akan dilakukan pada tiga sampel uji.
2. Pengujian Validasi digunakan untuk menguji seluruh kebutuhan apakah sistem dapat memenuhi spesifikasi fungsional sistem yang telah ditetapkan atau tidak. Pengujian validasi dilakukan dengan cara memeriksa apakah sistem sudah berjalan dengan baik dan tidak terjadi kesalahan. Pengujian validasi dilakukan pada semua kebutuhan sistem dengan *scenario-based testing* yang merupakan pengujian dari *black box testing*.

Dengan menganalisis dari hasil pengujian tersebut dapat mengetahui kekurangan apa saja yang terdapat di sistem yang telah dibuat dan juga mendapatkan apa saja yang harus

diperbaiki pada sistem. Analisis sistem untuk mengetahui apakah sistem telah sesuai dengan spesifikasi kebutuhan yang telah ditetapkan sebelumnya sehingga hasil pengujian akan menjadi pertimbangan sebagai kesimpulan atau evaluasi untuk membuat perbaikan sistem.

3.6 Kesimpulan

Tahap pengambilan kesimpulan dilakukan setelah melakukan pengujian dan analisis terhadap sistem. Kesimpulan diambil berdasarkan pada hasil dari pengujian dan analisis. Dengan adanya kesimpulan maka akan didapat inti dari hasil keseluruhan proses penelitian. Selain itu kesimpulan dapat memberikan saran untuk membangun sistem selanjutnya.



BAB 4 REKAYASA KEBUTUHAN

Rekayasa kebutuhan merupakan tahapan pertama dalam pengembangan sebuah sistem. Rekayasa kebutuhan bertujuan untuk menentukan kebutuhan apa saja yang harus terdapat pada sistem aplikasi manajemen bahan baku dan pemesanan pada sablon malang berdasarkan hasil observasi dan wawancara.

Rekayasa kebutuhan sistem juga menjelaskan siapa saja yang berperan dalam sistem. Selain itu, bagian ini juga dibuat pemodelan-pemodelan kebutuhan seperti *use case diagram* dan *use case scenario diagram* yang berguna untuk membantu dan meningkatkan pemahaman mengenai kebutuhan-kebutuhan yang ada pada sistem.

4.1 Elisitasi Kebutuhan

Kebutuhan aplikasi manajemen bahan baku dan pemesanan pada sablon malang didapat dari observasi dan wawancara pada sablon malang. Observasi yang dilakukan yaitu melihat langsung proses pembuatan pesanan didalam tempat produksi sablon dan konveksi. Penulis juga melihat tempat penyimpanan bahan baku yang terdapat di sablon malang. Hal itu dilakukan untuk mengetahui proses bisnis yang terjadi dalam proses pemesanan dan pencatatan bahan baku di sablon malang. Dalam proses observasi ini diperoleh data tentang proses bisnis yang terjadi di sablon malang. Sedangkan Wawancara menanyakan tentang proses produksi yang terjadi secara rinci kepada pemilik sablon dan permasalahan yang terdapat pada sablon malang tentang bahan baku. Serta wawancara tentang proses pemesanan yang ditujukan kepada kasir. Data yang diperoleh dari wawancara ini terdapat beberapa stakeholder yang berpengaruh dalam sablon malang yaitu pimpinan sablon malang, pimpinan produksi sablon dan konveksi, kasir, serta Gudang bahan baku.

Pada elisitasi kebutuhan menggunakan observasi didapat bahwa proses pembuatan pesanan terbagi menjadi 2 tahapan dan setiap tahapan berada ditempat yang berbeda dan memiliki penanggung jawab masing-masing. Tahapan pertama yaitu pembuatan pesanan ditempat konveksi, yang bertanggung jawab dalam tahap ini adalah satu orang yaitu pimpinan konveksi dan memiliki beberapa karyawan. Pada tahapan ini bertujuan untuk menjahit kain menjadi pakaian sesuai pesanan. Proses pertama yang dilakukan adalah dengan mendapatkan catatan pesanan dan file gambar yang sudah di kirim oleh kasir kepada pimpinan konveksi. Kemudian pimpinan konveksi mengkordinasikan kepada karyawan untuk membuat pesanan. Setelah proses pembuatan selesai konveksi mengirimkan pesanan ke tempat selanjutnya untuk dilakukan pembuatan pesanan tahap kedua. Pada tahap kedua yaitu ditempat sablon yang bertanggung jawab dalam tahap ini adalah satu orang yaitu pimpinan sablon dan memiliki beberapa karyawan. Pada tahapan ini bertujuan untuk membuat sablon selesai pesanan ke pakaian yang sudah dikirimkan oleh konveksi. Untuk proses dari tahap ini sama dengan tahap sebelumnya. Untuk bahan baku yang berada di sablon malang juga terbagi menjadi 2 tempat yaitu ditempat sablon dan ditempat konveksi. Pencatatan bahan baku dicatat oleh masing-masing pimpinan dengan menggunakan cara manual dan pengecekanya hanya satu bulan sekali. Pembelian bahan baku dibeli oleh 1 orang khusus yang bertugas membelikan bahan baku baik itu

sablon maupun konveksi, yang pelaksanaannya hanya ketika pimpinan mengintruksikan untuk membeli.

Narasumber dari wawancara yaitu pimpinan sablon malang dan kasir. Wawancara kepada pimpinan sablon menanyakan tentang proses pembuatan pesanan dari awal sampai selesai. Hasil dari wawancara tersebut didapat bahwa proses pemesanan terdiri dari beberapa tahap. Tahap pertama yaitu berada di tempat kasir yang berada di toko sablon malang didalamnya terdapat proses antara pelanggan dan kasir dalam membuat suatu pesanan. Kemudian catatan pesanan diberikan kepada masing-masing pimpinan untuk dibuat acuan pembuatan pesanan. Setelah semua proses pembuatan pesanan ditempat konveksi dan sablon selesai maka akan diantar kembali ke toko yang akan diterima oleh kasir untuk diberikan kepada pelanggan. Selain proses pembuatan pesanan peneliti juga menanyakan tentang permasalahan yang terjadi didalam sablon malang dan juga menanyakan tentang keinginan pemilik sablon terhadap sistem yang dibuat yaitu kemudahan dalam pemakaian aplikasi yang kebanyakan pemangku kepentingan masih belum handal dalam mengoperasikan sebuah sistem. Untuk wawancara kepada kasir peneliti menanyakan tentang proses pembuatan pesanan. Hasil dari wawancara tersebut yaitu data dari pesanan yang diberikan kepada masing-masing pimpinan dan pelanggan. Pada lampiran A akan dijelaskan bisnis proses yang terjadi didalam sablon malang.

4.1.1 Gambaran Umum Sistem

Berdasarkan permasalahan yang terdapat pada latar belakang, dan hasil dari observasi serta wawancara sistem harus dapat digunakan oleh pimpinan sablon malang, kasir, pimpinan produksi sablon, pimpinan produksi konveksi dan juga Gudang. Kasir dapat mencatat pemesanan sesuai dengan apa yang sudah ditentukan oleh pelanggan baik itu jadwal selesai maupun yang lainnya. Admin disini dibedakan menjadi dua yaitu admin konveksi dan admin sablon, dimana tiap admin tugasnya sama tetapi dalam ruang lingkup yang berbeda. Admin konveksi mengurus bagian konveksi sedangkan admin sablon mengurus bagian sablon. Kasir bertugas untuk membuat pesanan dan mencatat semua hal yang berkaitan tentang pemesanan. Gudang dapat melihat seluruh bahan baku yang tersedia dan menambahkan bahan baku yang baru didapatkan dan melakukan pembelian bahan baku. Seluruh user dapat melihat proses pembuatan suatu pesanan yang didalamnya berisi informasi sudah sampai di bagian apa proses pesanan tersebut. Sistem ini dirancang dengan menggunakan Bahasa pemrograman PHP digabungkan dengan *framework CI* dan bisa diakses melalui website.

4.2 Spesifikasi Kebutuhan

Proses ini melakukan identifikasi aktor yang akan menggunakan sistem dan kebutuhan yang didapat akan dibagi menjadi kebutuhan fungsional dan non-fungsional. kemudian dilakukan pemodelan kebutuhan yang berupa use case diagram dan scenario diagram.

4.2.1 Identifikasi Aktor

Pada tahapan ini akan diidentifikasi siapa saja aktor yang berinteraksi langsung terhadap sistem aplikasi ini. Pada table 4.1 akan diperlihatkan beberapa aktor dan penjelasannya.

Tabel 4.1. Tabel Aktor dan Penjelasanya

Aktor	Penjelasan
User	User merupakan aktor yang dapat menggunakan sistem. User adalah aktor yang hanya bisa melihat halaman depan dari sistem. Untuk bisa menjalankan sistem harus masuk terlebih sesuai status user.
Admin Konveksi	Admin Konveksi adalah pimpinan konveksi yang bertanggung jawab atas semua yang terjadi di konveksi dan yang mengatur serta mencatat setiap bahan baku beserta jadwal produksi yang terdapat dalam proses konveksi dan memverifikasi proses pesanan.
Admin Sablon	Admin Sablon adalah pimpinan sablon yang bertanggung jawab atas semua yang terjadi di sablon dan yang mengatur serta mencatat setiap bahan baku beserta jadwal produksi yang terdapat dalam proses sablon dan memverifikasi proses pesanan.
Kasir	Kasir adalah orang yang melayani langsung pelanggan. Kasir bertugas untuk membuat pesanan dan mencatat semua hal yang telah disetujui antara pelanggan dan kasir untuk proses pemesanan.
Gudang	Gudang adalah orang yang bertugas untuk membelikan bahan baku yang dibutuhkan untuk proses pemesanan. Gudang dapat menambahkan bahan baku yang baru didapatkan.
Admin	Admin adalah orang yang bertugas untuk mengelola seluruh sistem dan diberikan hak akses untuk mengelola user.

4.2.2 Analisis Kebutuhan Fungsional

Kebutuhan Fungsional merupakan suatu layanan yang harus disediakan oleh sistem, bagaimana sistem harus memberi reaksi terhadap data yang masuk, dan sistem harus berjalan dalam situasi yang ditentukan. Daftar kebutuhan fungsional ditunjukkan pada Table 4.2. setiap kebutuhan akan diberikan kode MKB_XX_YYYY. MKB menunjukan nama sistem sedangkan XX menunjukan nomor kebutuhan fungsional atau non fungsional dan YYYY merupakan nomor kebutuhan.

Tabel 4.2 Tabel Kebutuhan Fungsional

No	Kode	Kebutuhan	Use Case
1	MKB_01_0100	Sistem harus dapat menyediakan fungsi <i>login</i>	Login

		untuk user masuk kedalam sistem sesuai dengan status user	
1.1	MKB_01_0101	Data yang harus diisi adalah <i>username</i> dan <i>password</i>	
2	MKB_01_0200	Sistem harus dapat menyediakan fungsi <i>logout</i> untuk user keluar dari sistem menuju halaman awal	Logout
3	MKB_01_0300	Sistem harus dapat menyediakan layanan untuk kasir mencatat pesanan pelanggan	Membuat Pesanan
3.1	MKB_01_0301	Catatan pesanan yang harus diisi adalah nama pemesan, jenis pesanan, nama produk banyak pesanan, harga satuan, total harga, alamat pemesan, no HP pemesan, status, tanggal pemesanan, deadline pesanan, catatan.	
4	MKB_01_0400	Sistem harus dapat melihat daftar pesanan	Melihat Pesanan
5	MKB_01_0500	Sistem harus dapat menghapus daftar pesanan	Menghapus Pesanan
6	MKB_01_0600	Sistem harus dapat mengelola data pesanan	Mengubah Pesanan
7	MKB_01_0700	Sistem harus dapat mengunggah gambar pesanan	Mengunggah Gambar Pesanan
7.1	MKB_01_0701	Mengunggah gambar dilakukan saat mencatat pesanan dan hanya ekstensi jpg, jpeg, png atau ekstensi gambar yang lain yang bisa diunggah	
8	MKB_01_0800	Sistem harus dapat melihat daftar bahan baku konveksi	Melihat Bahan Konveksi

9	MKB_01_0900	Sistem harus dapat melihat daftar bahan baku sablon	Melihat Bahan sablon
10	MKB_01_1000	Sistem harus dapat Menambahkan Bahan Baku Baru konveksi dan sablon	Menambahkan Bahan Baku
10.1	MKB_01_1001	-	
11	MKB_01_1100	Sistem harus dapat melihat daftar bahan baku konveksi dan sablon	Melihat Bahan Konveksi dan Sablon
12	MKB_01_1200	Sistem harus dapat menghapus bahan baku konveksi dan sablon	Menghapus Bahan Konveksi dan Sablon
13	MKB_01_1300	Sistem harus dapat mengelola data bahan baku konveksi dan sablon	Mengubah Bahan Konveksi dan Sablon
14	MKB_01_1400	Sistem harus dapat mengunggah gambar bahan baku	Mengunggah Gambar Bahan Baku
14.1	MKB_01_1401	Mengunggah gambar dilakukan saat mencatat pesanan dan hanya ekstensi jpg, jpeg, png atau ekstensi gambar yang lain yang bisa diunggah	
15	MKB_01_1500	Sistem harus dapat melakukan penambahan jumlah bahan baku	Membuat Pembelian Bahan Baku
15.1	MKB_01_1501	-	
16	MKB_01_1600	Sistem harus dapat mengelola data pembelian Bahan Baku	Mengubah Pembelian Bahan Baku
17	MKB_01_1700	Sistem harus dapat melihat daftar pembelian bahan baku	Melihat Pembelian Bahan Baku
18	MKB_01_1800	Sistem harus dapat menghapus pembelian bahan baku	Menghapus Pembelian Bahan Baku

19	MKB_01_1900	Sistem harus dapat membuat perencanaan produksi setiap periode	Membuat Jadwal Induk Produksi
19.1	MKB_01_1901	Data yang harus diisi adalah forcast dan mps	
20	MKB_01_2000	Sistem harus dapat menampilkan rencana produksi setiap periode	Melihat Jadwal Induk Produksi
20.1	MKB_01_2001	Data yang ditampilkan akan diolah dari dari forcast dan mps yang sudah dimasukkan sesuai rencana	
21	MKB_01_2100	Sistem dapat mengelola data jadwal induk produksi	Mengubah Jadwal Induk Produksi
22	MKB_01_2200	Sistem dapat Menghapus data jadwal induk produksi	Menghapus Jadwal Induk Produksi
23	MKB_01_2300	Sistem harus bisa membuat Request Order pembelian Bahan Baku	Membuat Request Order
23.1	MKB_01_2301	-	
24	MKB_01_2400	Sistem harus bisa mengelola data Request Order	Mengubah Request Order
25	MKB_01_2500	Sistem Harus dapat menghapus data Request Order	Menghapus Request Order
26	MKB_01_2600	Sistem harus dapat melihat semua data Request Order	Melihat Request Order
27	MKB_01_2700	Sistem harus dapat menerima dan menolak Request Order	Memvalidasi Request Order
28	MKB_01_2800	Sistem harus dapat mengecek ketersediaan Bahan Baku yang ada dengan Pesanan	Mengecek Ketersediaan
29	MKB_01_2900	Sistem harus dapat mengupdate status pada pesanan	Update Status Pesanan

30	MKB_01_3000	Sistem harus dapat mencetak Nota sesuai data	Mencetak Nota
31	MKB_01_3100	Sistem harus dapat Membuat catatan bahan baku yang sudah dipakai untuk pesanan	Membuat Catatan pesanan
31.1	MKB_01_3101	-	
32	MKB_01_3200	Sistem harus dapat melihat daftar catatan pesanan	Melihat Catatan Pesanan
33	MKB_01_3300	Sistem harus bisa menambahkan user yang memakai sistem	Menambah User
33.1	MKB_01_3301	Data yang diisi adalah Nama User, username dan password	
34	MKB_01_3400	Sistem harus bisa mengelola data user	Mengubah User
35	MKB_01_3500	Sistem Harus bisa menghapus data user	Menghapus User
36	MKB_01_3600	Sistem harus bisa melihat semua data user	Melihat User

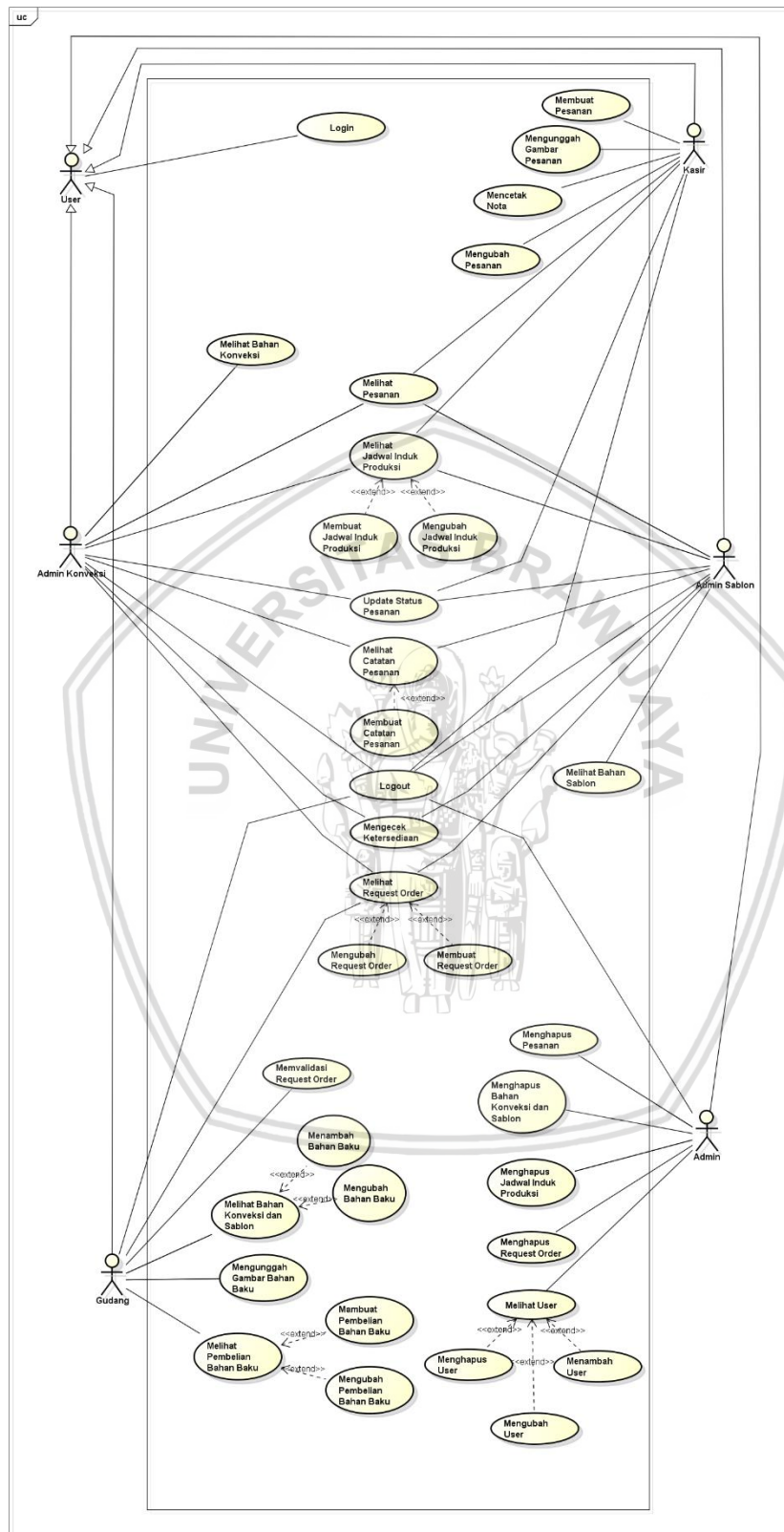
4.2.3 Analisis Kebutuhan Non Fungsional

Kebutuhan Non Fungsional kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem. Daftar kebutuhan non fungsional ditunjukkan pada Table 4.3. setiap kebutuhan akan diberikan kode MKB_XX_YYYY. MKB menunjukkan nama sistem sedangkan XX menunjukkan nomor kebutuhan fungsional atau non fungsional dan YYYY merupakan nomor kebutuhan.

Tabel 4.3 Tabel Kebutuhan Non Fungsional

Kode	Parameter	Deskripsi
MKB_02_1000	<i>Compatibility</i>	<p>Sistem dapat berjalan pada beberapa jenis peramban yakni:</p> <ul style="list-style-type: none"> - Chrome 65 - Firefox 61 - Opera 51

4.2.4 Use Case Diagram



Gambar 4.1 Use Case Sistem

Diagram *Use Case* dari sistem ini dibuat berdasarkan daftar kebutuhan fungsional yang telah dibuat dan akan digambarkan pada gambar 4.1 diatas ini. *Use case* dalam sistem ini memiliki 5 aktor yang berawal dari user. Untuk masuk kedalam sistem user harus melakukan *login* terlebih dahulu dan akan masuk sesuai dengan status user. Admin konveksi dan admin sablon memiliki *use case* yang sama, cuma perbedaannya adalah database kedua aktor tersebut karena memiliki tempat sendiri. Gudang mempunyai akses untuk mengelola bahan baku dari sablon dan konveksi dan membuat validasi dari request order sedangkan kasir memiliki *use case* membuat pesanan dan mengunggah gambar pesanan, membuat nota. Admin disini bisa melakukan apapun yang dilakukan oleh user, hanya admin yang bisa untuk menghapus data dan melakukan pengelolaan user. *Use case* yang dimiliki semua actor yaitu dan *logout*.

4.2.5 Use Case Scenario

Skenario *use case* dari sistem manajemen bahan baku dan pemesanan ini akan dijelaskan dibawah ini.

4.2.5.1 Login

Login merupakan fungsi untuk memverifikasi user dengan memasukkan *username* dan *password* agar bisa masuk kedalam sistem sebagai pengguna baik itu admin konveksi, admin sablon, kasir, dan Gudang. Skenario diagram *login* akan dijelaskan pada table 4.4.

Table 4.4 Skenario Use Case Login

Login	
Actor	User
Objective	Mengizinkan user untuk masuk kedalam sistem sebagai admin konveksi, admin sablon, kasir, dan Gudang
Pre-Condition	Membuka halaman awal sistem
Main Flow	<ol style="list-style-type: none"> 1. User memasukkan <i>username</i> dan <i>password</i> lalu memilih tombol login 2. Sistem akan menampilkan halaman awal sesuai dengan status akun user yaitu sebagai kasir, admin konveksi, admin sablon, Gudang
Alternative Flow	<ol style="list-style-type: none"> 1. Jika saat menekan tombol login <i>username</i> atau <i>password</i> salah maka sistem akan menampilkan pesan "<i>username</i> dan <i>password</i> salah" 2. Jika saat menekan tombol login <i>username</i> atau <i>password</i> kosong maka sistem akan menampilkan pesan "<i>username</i> dan <i>password</i> tidak boleh kosong"
Post-Condition	User dapat masuk kedalam sistem sesuai status user dan masuk ke halaman utama

4.2.5.2 Logout

Logout merupakan fungsi untuk pengguna keluar dari sistem menuju halaman awal. Skenario diagram *logout* akan dijelaskan pada table 4.5.

Table 4.5 Skenario Use Case Logout

Logout	
Actor	Admin Konveksi, Admin Sablon, Gudang, Kasir
Objective	Admin Konveksi, Admin Sablon, Gudang, Kasir keluar dari sistem menuju halaman awal
Pre-Condition	Admin Konveksi, Admin Sablon, Gudang, Kasir sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Admin Konveksi, Admin Sablon, Gudang, Kasir memilih menu Logout 2. Sistem akan mengembalikan ke status user dan menampilkan halaman awal
Alternative Flow	-
Post-Condition	Admin Konveksi, Admin Sablon, Gudang, Kasir, Admin berhasil keluar dari sistem dan kembali ke halamana awal

4.2.5.3 Membuat Pesanan

Membuat Pesanan merupakan fungsi untuk kasir mencatat pesanan pelanggan yang baru akan memesan. Dengan memasukkan data yang sesuai dengan pesanan maka kasir akan memasukkan hasilnya ke dalam catatan yang nanti bisa dilihat oleh seluruh pengguna. Skenario diagram membuat pesanan akan dijelaskan pada table 4.6.

Table 4.6 Skenario Use Case Membuat Pesanan

Membuat Pesanan	
Actor	Kasir
Objective	Kasir mencatat semua kebutuhan dan keinginan pelanggan kedalam form
Pre-Condition	<ol style="list-style-type: none"> 1. Kasir sudah berada didalam menu tambah pesanan 2. Sistem telah menampilkan form pesanan
Main Flow	<ol style="list-style-type: none"> 1. Kasir mengisi form pesanan yang berisi nama pemesan, jenis pesanan, banyak pesanan, nama produk, harga satuan, total harga, alamat pemesan, no HP pemesan, status, tanggal pemesanan, deadline pesanan, catatan, pembayaran, tombol upload gambar pesanan, jadwal produksi 2. Sistem menyimpan form pesanan kedalam database

Alternative Flow	1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Kasir telah berhasil menyimpan pesanan dan mengembalikan ke dalam daftar pesanan

4.2.5.4 Mengunggah Gambar Pesanan

Mengunggah Gambar Pesanan merupakan fungsi untuk mengunggah gambar pesanan yang telah diberi oleh pelanggan dan nanti akan dapat dilihat oleh seluruh pengguna untuk sebagai acuan membuat pesanan. Skenario diagram mengunggah gambar akan dijelaskan pada table 4.7

Table 4.7 Skenario Use Case Mengunggah Gambar Pesanan

Mengunggah Gambar	
Actor	Kasir
Objective	Kasir mengunggah gambar yang telah diberikan oleh pelanggan
Pre-Condition	Kasir sudah didalam menu membuat pemesanan
Main Flow	<ol style="list-style-type: none"> 1. Kasir memilih fungsi unggah pada saat melakukan pengisian pada form pesanan 2. Sistem akan menampilkan daftar gambar didalam database komputer 3. Kasir memilih gambar yang akan di unggah, ekstensi gambar bisa berupa jpeg, jpg, png ataupun ekstensi gambar lainnya 4. Sistem menyimpan gambar kedalam database
Alternative Flow	1. Jika gambar tidak sesuai persyaratan maka sistem akan menampilkan pesan “Gambar tidak sesuai format”
Post-Condition	Kasir telah berhasil mengunggah gambar

4.2.5.5 Mencetak Nota

Membuat Catatan Pesanan Mencetak Nota merupakan fungsi untuk Mencetak data Nota yang berada didalam database. Skenario diagram Mencetak Nota akan dijelaskan pada table 4.8.

Table 4.8 Skenario Use Case Mencetak Nota

Mencetak Nota	
Actor	Kasir
Objective	Kasir Mencetak data Nota sesuai data pesanan di dalam database

Pre-Condition	<ol style="list-style-type: none"> 1. Kasir sudah berada didalam sistem 2. Data Nota sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Kasir memilih menu Daftar Pesanan 2. Sistem akan menampilkan Semua daftar Pesanan yang didalamnya terdapat informasi Pesanan 3. Kasir memilih fungsi tombol Cetak pada salah satu Pesanan 4. Sistem Mencetak data Nota dari database
Alternative Flow	-
Post-Condition	Kasir berhasil Mencetak data Nota dari database

4.2.5.6 Mengubah Pesanan

Mengubah Pesanan merupakan fungsi untuk mengganti data pesanan dengan yang baru atau menghapusnya dari daftar pesanan. Skenario diagram mengubah pesanan akan dijelaskan pada table 4.9.

Table 4.9 Skenario Use Case Mengubah Pesanan

Mengubah Pesanan	
Actor	Kasir
Objective	Kasir mengubah data pesanan yang sudah berada di daftar pesanan
Pre-Condition	<ol style="list-style-type: none"> 1. Kasir sudah berada di dalam menu daftar pesanan 2. Pesanan sudah terdaftar kedalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Kasir memilih fungsi tombol ubah pada salah satu daftar pesanan 3. Sistem menampilkan form pesanan yang dipilih berisi nama pemesan, jenis pesanan, nama produk, banyak pesanan, harga satuan, total harga, alamat pemesan, no HP pemesan, status, tanggal pemesanan, deadline pesanan, catatan, pembayaran, tombol upload gambar pesanan, jadwal produksi 2. Kasir mengubah form bahan baku sesuai keinginan
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Kasir telah berhasil mengubah data pesanan

4.2.5.7 Melihat Pesanan

Melihat Pesanan merupakan fungsi untuk melihat semua daftar pesanan yang berada pada database. Skenario diagram melihat pesanan akan dijelaskan pada table 4.10.

Table 4.10 Skenario Use Case Melihat Pesanan

Melihat Pesanan	
Actor	Admin Sablon, Admin Konveksi, atau Kasir
Objective	Admin Sablon, Admin Konveksi, atau Kasir melihat semua daftar pesanan
Pre-Condition	Admin Sablon, Admin Konveksi, atau Kasir sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Admin Sablon, Admin Konveksi, atau Kasir memilih menu daftar Pesanan 2. Sistem akan menampilkan semua daftar Pesanan yang didalamnya terdapat informasi pesanan
Alternative Flow	-
Post-Condition	Admin Sablon, Admin Konveksi, atau Kasir telah berhasil melihat semua daftar pesanan

4.2.5.8 Melihat Jadwal Produksi

Melihat Jadwal Induk Produksi merupakan fungsi untuk melihat laporan jadwal induk produksi yang sudah diolah oleh sistem. Skenario diagram Melihat Jadwal Induk Produksi akan dijelaskan pada table 4.11.

Table 4.11 Skenario Use Case Melihat Jadwal Induk Produksi

Melihat Jadwal Induk Produksi	
Actor	Kasir, Admin Konveksi dan Admin Sablon
Objective	Kasir, Admin Konveksi dan Admin Sablon melihat Daftar jadwal induk produksi yang sudah diolah oleh sistem
Pre-Condition	Kasir, Admin Konveksi dan Admin Sablon sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Kasir, Admin Konveksi dan Admin Sablon memilih menu Jadwal Produksi 2. Sistem akan menampilkan Semua Laporan yang didalamnya terdapat informasi jadwal induk produksi
Alternative Flow	-
Post-Condition	Kasir, Admin Konveksi dan Admin Sablon berhasil melihat semua Daftar Jadwal Induk Produksi

4.2.5.9 Update Status Pesanan

Update Status Pesanan merupakan fungsi untuk Mengupdate pesanan sesuai proses pesanan saat itu. Skenario diagram Update Status Pesanan akan dijelaskan pada table 4.12.

Table 4.12 Skenario Use Case Update Status Pesanan

Update Status Pesanan	
Actor	Kasir, Admin Sablon dan Admin Konveksi
Objective	Kasir, Admin Sablon dan Admin Konveksi update pesanan sesuai keadaan pesanan saat itu
Pre-Condition	<ol style="list-style-type: none"> 1. Kasir, Admin Sablon atau Admin Konveksi sudah berada didalam sistem 2. Data Pesanan sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Admin Sablon atau Admin Konveksi memilih menu Jadwal Produksi. Kasir memilih menu Daftar Pesanan 2. Sistem akan menampilkan semua Daftar Produksi dan Daftar Pesanan yang sudah ada pada database 3. Admin Sablon atau Admin Konveksi memilih fungsi tombol Pesanan kemudian detail. Kasir memilih fungsi tombol edit 4. Sistem menampilkan form detail pesanan yang dipilih kemudian menampilkan form selanjutnya sesuai pilihan status 5. Kasir, Admin Sablon atau Admin Konveksi mengisi status pesanan
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong” 2. Jika Status Pesanan yaitu “Check Ketersediaan” maka akan menampilkan form Chek Ketersediaan 3. Jika Status Pesanan yaitu “Proses Selesai” maka akan menampilkan form catatan pesanan
Post-Condition	Kasir, Admin Sablon atau Admin Konveksi berhasil mengupdate status pesanan

4.2.5.10 Membuat Catatan Pesanan

Membuat Catatan Pesnan merupakan fungsi untuk mencatat bahan baku yang habis dipakai saat proses pembuatan pesanan selesai. Skenario diagram mencatat hasil pesanan akan dijelaskan pada table 4.13.

Table 4.13 Skenario Use Case Mencatat Hasil Pesanan

Mencatat Hasil Pesanan	
Actor	Admin Sablon atau Admin Konveksi
Objective	Admin Sablon, Admin Konveksi mencatat semua bahan baku yang dipakai saat proses pesanan
Pre-Condition	<ol style="list-style-type: none"> 1. Admin Sablon atau Admin Konveksi sudah berada didalam sistem 2. Status Pesanan berupa Proses

Main Flow	<ol style="list-style-type: none"> 1. Admin Sablon atau Admin Konveksi memilih menu Jadwal Produksi 2. Sistem akan menampilkan Daftar Produksi yang berisi informasi Produksi 3. Admin Sablon atau Admin Konveksi memilih salah satu daftar dan menekan tombol pesanan 4. Sistem akan menampilkan daftar pesanan yang sesuai jadwal produksi tersebut 5. Admin Sablon atau Admin Konveksi memilih salah satu pesanan yang status pesanan berupa proses dan menekan tombol detail 6. Sistem akan menampilkan detail pesanan tersebut 7. Admin Sablon atau Admin Konveksi menekan tombol Tambah Catatan 8. Sistem menampilkan form catatan pesanan yang berisi tanggal catatan, nama bahan baku, jumlah, dan catatan 9. Admin Sablon atau Admin Konveksi mengisi form catatan pesanan 10. Admin Sablon atau Admin Konveksi memilih fungsi selesai untuk melanjutkan ke dalam menu daftar catatan pesanan
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong” 2. Jika ingin menambahkan lebih dari satu bahan baku kedalam form maka Admin Sablon, Admin Konveksi, atau Kasir memilih fungsi tambah
Post-Condition	Admin Sablon atau Admin Konveksi berhasil mencatat semua bahan baku yang telah dipakai saat proses pemesanan

4.2.5.11 Melihat Catatan Pesanan

Melihat Catatan Pesanan merupakan fungsi untuk melihat semua daftar catatan pesanan yang telah dibuat. Skenario diagram Melihat Catatan Pesanan akan dijelaskan pada table 4.14.

Table 4.14 Skenario Use Case Melihat Catatan Pesanan

Catatan Pesanan	
Actor	Kasir, Admin Konveksi, atau Admin Sablon
Objective	Kasir, Admin Konveksi, atau Admin Sablon melihat daftar catatan yang telah dibuat

Pre-Condition	Admin Konveksi, atau Admin Sablon sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Admin Konveksi, atau Admin Sablon memilih menu Daftar Catatan 2. Sistem akan menampilkan semua daftar catatan yang telah dibuat
Alternative Flow	-
Post-Condition	Admin Konveksi, atau Admin Sablon telah berhasil melihat semua daftar catatan

4.2.5.12 Membuat Jadwal Induk Produksi

Membuat Jadwal Induk Produksi merupakan fungsi untuk membuat perencanaan dalam produksi setiap periode. Skenario diagram Membuat Jadwal Induk Produksi akan dijelaskan pada table 4.15.

Table 4.15 Skenario Use Case Membuat Jadwal Induk Produksi

Membuat Jadwal Induk Produksi	
Actor	Admin Konveksi, kasir, atau Admin Sablon
Objective	Admin Konveksi, kasir, atau Admin Sablon membuat jadwal produksi produksi
Pre-Condition	<ol style="list-style-type: none"> 1. Admin Konveksi, kasir, atau Admin Sablon sudah berada didalam menu Tambah Jadwal 2. Sistem telah menampilkan form jadwal induk produksi
Main Flow	<ol style="list-style-type: none"> 1. Sistem menampilkan form jadwal induk produksi 2. Admin Konveksi, kasir, atau Admin Sablon mengisi form jadwal induk produksi yang berisi tanggal produksi, tanggal selesai dan mps 3. Admin Konveksi, kasir, atau Admin Sablon memilih fungsi Submit untuk melanjutkan ke dalam menu daftar jadwal produksi
Alternative Flow	Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Admin Konveksi, kasir, atau Admin Sablon berhasil membuat Jadwal Induk Produksi

4.2.5.13 Mengubah Jadwal Induk Produksi

Mengubah Jadwal Induk Produksi merupakan fungsi untuk Mengubah laporan jadwal induk produksi yang sudah diolah oleh sistem. Skenario diagram Mengubah Jadwal Induk Produksi akan dijelaskan pada table 4.16.

Table 4.16 Skenario Use Case Mengubah Jadwal Induk Produksi

Mengubah Jadwal Induk Produksi	
Actor	Admin Konveksi, kasir, atau Admin Sablon
Objective	Admin Konveksi, kasir, atau Admin Sablon mengubah jadwal induk produksi yang sudah diolah oleh sistem dengan yang baru
Pre-Condition	<ol style="list-style-type: none"> 1. Admin Konveksi, kasir, atau Admin Sablon sudah berada didalam system 2. Data jadwal induk produksi sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Admin Konveksi, kasir, atau Admin Sablon memilih menu Daftar Jadwal Produksi 2. Sistem akan menampilkan Laporan yang didalamnya terdapat informasi jadwal induk produksi 3. Admin Konveksi, kasir, atau Admin Sablon memilih fungsi tombol edit 4. Sistem menampilkan form jadwal induk produksi yang dipilih berisi tanggal produksi, tanggal selesai dan mps 5. Admin Konveksi, kasir, atau Admin Sablon mengubah form jadwal induk produksi sesuai keinginan dan menekan tombol submit
Alternative Flow	Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Admin Konveksi, kasir, atau Admin Sablon berhasil mengubah data Jadwal Induk Produksi dengan yang baru

4.2.5.14 Mengecek Ketersediaan

Mengecek Ketersediaan merupakan fungsi untuk mengecek bahan baku yang digunakan sudah sesuai dengan kebutuhan pesanan. Skenario diagram Mengecek Ketersediaan akan dijelaskan pada table 4.17.

Table 4.17 Skenario Use Case Mengecek Ketersediaan

Mengecek Ketersediaan	
Actor	Admin Sablon atau Admin Konveksi
Objective	Admin Sablon atau Admin Konveksi Mengecek Ketersediaan antara Kebutuhan pesanan dengan Jumlah Bahan Baku yang ada
Pre-Condition	<ol style="list-style-type: none"> 1. Admin Sablon atau Admin Konveksi sudah berada didalam menu Jadwal Produksi 2. Sistem telah menampilkan daftar produksi
Main Flow	<ol style="list-style-type: none"> 1. Admin Sablon atau Admin Konveksi memilih salah satu daftar dan menekan tombol pesanan 2. Sistem akan menampilkan daftar pesanan yang sesuai jadwal produksi tersebut

	<ol style="list-style-type: none"> Admin Sablon atau Admin Konveksi memilih salah satu pesanan yang status pesanan berupa diantrikan dan menekan tombol detail Sistem akan menampilkan detail pesanan tersebut Admin Sablon atau Admin Konveksi menekan tombol Tambah Catatan Sistem menampilkan form catatan pesanan yang berisi tanggal catatan, nama bahan baku, jumlah, dan catatan Admin Sablon atau Admin Konveksi mengisi form catatan pesanan Admin Sablon atau Admin Konveksi memilih fungsi selesai untuk melanjutkan ke dalam menu daftar catatan pesanan
Alternative Flow	<ol style="list-style-type: none"> Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong” Jika Bahan Baku mencukupi maka status pesanan bisa dilanjutkan Jika Bahan Baku tidak mencukupi maka tidak dapat melanjutkan pesanan dan mengeluarkan tulisan “Bahan Baku Tidak Mencukupi”
Post-Condition	Admin Sablon atau Admin Konveksi berhasil mengupdate Mengecek Ketersediaan

4.2.5.15 Membuat Request Order

Membuat Request Order merupakan fungsi untuk membuat permintaan pembelian bahan baku kepada Gudang. Skenario diagram Membuat Request Order akan dijelaskan pada table 4.18.

Table 4.18 Skenario Use Case Membuat Request Order

Membuat Request Order	
Actor	Admin Sablon atau Admin Konveksi
Objective	Admin Sablon atau Admin Konveksi membuat Request Order untuk permintaan pembelian terhadap Gudang
Pre-Condition	<ol style="list-style-type: none"> Admin Sablon atau Admin Konveksi sudah berada didalam menu Tambah Request Order Sistem telah menampilkan form request order
Main Flow	<ol style="list-style-type: none"> Admin Sablon dan Konveksi mengisi form Request Order yang berisi Tanggal, Catatan, Nama Bahan Baku, Jumlah Bahan Baku Admin Sablon dan Konveksi memilih fungsi selesai untuk melanjutkan ke dalam menu daftar Request Order

Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong” 2. Jika ingin menambah lebih dari satu maka menekan tombol Tambah Bahan 3. Jika ingin menghapus bahan yang sudah di isi maka menekan tombol Hapus Bahan
Post-Condition	Admin Sablon atau Admin Konveksi berhasil membuat status persediaan

4.2.5.16 Mengubah Request Order

Mengubah Request Order merupakan fungsi untuk Mengubah Request Order yang sudah diolah oleh sistem. Skenario diagram Mengubah Request Order akan dijelaskan pada table 4.19.

Table 4.19Skenario Use Case Mengubah Request Order

Mengubah Request Order	
Actor	Admin Sablon atau Admin Konveksi
Objective	Admin Sablon atau Admin Konveksi mengubah Request Order yang sudah berada didalam database
Pre-Condition	<ol style="list-style-type: none"> 1. Admin Sablon atau Admin Konveksi sudah didalam sistem 2. Data Request Order sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Admin Sablon atau Admin Konveksi memilih menu Request Order 2. Sistem akan menampilkan Daftar Request Order yang ada di database 3. Admin Sablon atau Admin Konveksi memilih fungsi tombol edit 4. Sistem menampilkan form Request Order yang dipilih berisi Tanggal, catatan, Nama Bahan Baku dan Jumlah Bahan Baku 5. Admin Sablon dan Admin Konveksi mengubah form Request Order sesuai keinginan 6. Admin Sablon dan Admin Konveksi memilih Tombol Submit untuk menyimpan
Alternative Flow	Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Admin Sablon dan Admin Konveksi berhasil mengubah data Request Order dengan yang baru

4.2.5.17 Melihat Request Order

Melihat Request Order merupakan fungsi untuk melihat Request Order yang sudah diolah oleh sistem. Skenario diagram Melihat Request Order akan dijelaskan pada table 4.20.

Table 4.20 Skenario Use Case Melihat Request Order

Melihat Request Order	
Actor	Gudang, Admin Sablon, atau Admin Konveksi
Objective	Gudang, Admin Sablon, atau Admin Konveksi melihat Request Order yang sudah diolah oleh sistem
Pre-Condition	Gudang, Admin Sablon, atau Admin Konveksi sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Gudang, Admin Sablon, atau Admin Konveksi memilih menu Request Order 2. Sistem akan menampilkan Semua Daftar Request Order yang berada di database
Alternative Flow	-
Post-Condition	Gudang, Admin Sablon, atau Admin Konveksi berhasil melihat semua Daftar Request Order

4.2.5.18 Melihat Bahan Konveksi

Melihat Bahan Konveksi merupakan fungsi untuk melihat semua daftar persediaan bahan baku yang berada pada konveksi. Skenario diagram Melihat Bahan Konveksi akan dijelaskan pada table 4.21.

Table 4.21 Skenario Use Case Melihat Bahan Konveksi

Melihat Bahan Konveksi	
Actor	Admin Konveksi
Objective	Admin Konveksi melihat semua daftar bahan baku yang berada pada konveksi
Pre-Condition	Admin Konveksi Masuk sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Admin Konveksi memilih menu Persediaan 2. Sistem akan menampilkan semua daftar persediaan bahan baku yang berada pada konveksi
Alternative Flow	-
Post-Condition	Admin Konveksi berhasil melihat semua daftar bahan baku yang berada pada konveksi

4.2.5.19 Melihat Bahan Sablon

Melihat Bahan Sablon merupakan fungsi untuk melihat semua daftar persediaan bahan baku yang berada pada sablon. Skenario diagram Melihat Bahan Sablon akan dijelaskan pada table 4.22.

Table 4.22 Skenario Use Case Melihat Bahan Konveksi

Melihat Bahan Sablon	
Actor	Admin Sablon
Objective	Admin Sablon melihat semua daftar bahan baku yang berada pada sablon
Pre-Condition	Admin Sablon sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Admin Konveksi memilih menu Persediaan 2. Sistem akan menampilkan semua daftar persediaan bahan baku yang berada pada konveksi
Alternative Flow	-
Post-Condition	Admin Sablon berhasil melihat semua daftar bahan baku yang berada pada sablon

4.2.5.20 Memvalidasi Request Order

Memvalidasi Request Order merupakan fungsi untuk Memvalidasi Request Order yang sudah diminta oleh admin sablon maupun admin konveksi. Skenario diagram Memvalidasi Request Order akan dijelaskan pada table 4.23.

Table 4.23 Skenario Use Case Memvalidasi Request Order

Memvalidasi Request Order	
Actor	Gudang
Objective	Gudang Memvalidasi Request Order yang berada didalam database
Pre-Condition	<ol style="list-style-type: none"> 1. Gudang sudah berada didalam sistem 2. Data Request Order sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Gudang memilih menu Request Order 2. Sistem akan menampilkan Semua Daftar Request Order 3. Gudang memilih fungsi tombol Validasi pada salah satu Data Request Order ditolak atau disetujui 4. Sistem memvalidasi Request Order dan melanjutkan hasil validasi ke admin sablon atau admin konveksi
Alternative Flow	-
Post-Condition	Gudang berhasil Memvalidasi Request Order dari database

4.2.5.21 Menambahkan Bahan Baku

Menambahkan Bahan Baku merupakan fungsi untuk Gudang dapat menambahkan bahan baku baru baik itu sablon maupun konveksi yang nanti akan masuk kedalam daftar bahan baku. Skenario diagram menambahkan bahan Baku akan dijelaskan pada table 4.24.

Table 4.24 Skenario Use Case Menambahkan Bahan Baku

Menambahkan Bahan Baku	
Actor	Gudang
Objective	Gudang menambahkan bahan baku baru baik konveksi maupun sablon
Pre-Condition	<ol style="list-style-type: none"> 1. Gudang sudah berada didalam menu Tambah Bahan baku 2. Sistem telah menampilkan form bahan baku
Main Flow	<ol style="list-style-type: none"> 1. Gudang mengisi form bahan baku yang berisi nama barang, jenis barang apakah milik konveksi atau milik sablon dan catatan yang bisa ditambahkan. 2. Gudang memilih tombol Submit untuk menyimpan bahan baku yang sudah dimasukkan 3. Sistem menyimpan bahan baku baru kedalam database sesuai jenis yang dipilih
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Gudang berhasil memasukkan bahan baku baru ke dalam sistem

4.2.5.22 Melihat Bahan Konveksi dan Sablon

Melihat Bahan Konveksi dan Sablon merupakan fungsi untuk melihat semua daftar bahan baku yang berada pada konveksi maupun Sablon. Skenario diagram Melihat Bahan Konveksi dan Sablon akan dijelaskan pada table 4.25.

Table 4.25 Skenario Use Case Melihat Bahan Konveksi dan Sablon

Melihat Bahan Konveksi dan Sablon	
Actor	Gudang
Objective	Gudang melihat semua daftar bahan baku yang berada pada konveksi maupun sablon
Pre-Condition	Gudang sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Gudang memilih menu Daftar bahan baku 2. Sistem akan menampilkan semua Daftar Bahan Baku yang berada pada Konveksi maupun Sablon
Alternative Flow	-

Post-Condition	Gudang berhasil melihat semua daftar bahan baku yang berada pada Konveksi maupun Sablon
----------------	---

4.2.5.23 Mengubah Bahan Sablon dan Konveksi

Mengubah bahan Sablon dan Konveksi merupakan fungsi untuk Gudang mengubah daftar bahan baku baik itu sablon maupun konveksi dengan yang baru atau menghapusnya dari daftar bahan baku. Skenario diagram mengubah bahan sablon dan konveksi akan dijelaskan pada table 4.26.

Table 4.26 Skenario Use Case Mengubah Bahan Sablon dan Konveksi

Mengubah Bahan Sablon dan Konveksi	
Actor	Gudang
Objective	Gudang mengubah bahan baku yang sudah berada pada database konveksi dan sablon
Pre-Condition	<ol style="list-style-type: none"> 1. Gudang sudah berada didalam sistem 2. Data Bahan telah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Gudang memilih menu daftar bahan baku 2. Sistem akan menampilkan semua daftar bahan baku 3. Gudang memilih fungsi tombol Edit bahan baku 4. Sistem menampilkan form bahan baku yang dipilih berisi nama bahan baku, dan catatan 5. Gudang mengubah form bahan baku sesuai keinginan
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Gudang berhasil mengubah bahan baku menjadi data yang baru ke dalam sistem

4.2.5.24 Mengunggah Gambar Bahan Baku

Mengunggah Gambar Bahan Baku merupakan fungsi untuk mengunggah gambar Bahan Baku sesuai dengan nama bahan. Skenario diagram mengunggah gambar Bahan Baku akan dijelaskan pada table 4.27.

Table 4.27 Skenario Use Case Mengunggah Gambar Bahan Baku

Mengunggah Gambar	
Actor	Gudang
Objective	Gudang mengunggah gambar yang telah diberikan oleh pelanggan
Pre-Condition	Gudang sudah berada didalam menu mencatat pemesanan

Main Flow	<ol style="list-style-type: none"> 1. Gudang memilih fungsi unggah pada saat melakukan pengisian pada form pesanan 2. Sistem akan menampilkan daftar gambar didalam database komputer 3. Gudang memilih gambar yang akan di unggah, ekstensi gambar bisa berupa jpeg, jpg, png ataupun ekstensi gambar lainnya 4. Sistem menyimpan gambar kedalam database
Alternative Flow	<ol style="list-style-type: none"> 1. Jika gambar tidak sesuai persyaratan maka sistem akan menampilkan pesan “Gambar tidak sesuai format”
Post-Condition	Gudang telah berhasil mengunggah gambar

4.2.5.25 Membuat Pembelian Bahan Baku

Membuat Pembelian Bahan Baku merupakan fungsi untuk Gudang dapat membuat pembelian bahan baku untuk dimasukkan ke dalam persediaan bahan baku. Skenario diagram menambahkan bahan Baku akan dijelaskan pada table 4.28.

Table 4.28 Skenario Use Case Membuat Pembelian Bahan Baku

Membuat Pembelian Bahan Baku	
Actor	Gudang
Objective	Gudang membuat pembelian bahan baku yang nanti otomatis menambahkan jumlah bahan baku
Pre-Condition	<ol style="list-style-type: none"> 1. Gudang sudah berada didalam menu Tambah Pembelian 2. Sistem menampilkan form pembelian
Main Flow	<ol style="list-style-type: none"> 1. Gudang mengisi form catatan pesanan yang berisi tanggal pembelian, nama bahan baku, jumlah, dan catatan 2. Gudang memilih fungsi Submit untuk melanjutkan ke dalam menu daftar catatan pesanan
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong” 2. Jika ingin menambahkan lebih dari satu bahan baku kedalam form maka memilih tombol Tambah Baan 3. Jika ingin menghapus Bahan maka memilih tombol Hapus Bahan
Post-Condition	Gudang berhasil menambahkan Pembelian ke dalam database

4.2.5.26 Melihat Pembelian Bahan Baku

Melihat Pembelian Bahan Baku merupakan fungsi untuk melihat semua daftar pembelian bahan baku yang sudah pernah dimasukkan. Skenario diagram Melihat Pembelian Bahan Baku akan dijelaskan pada table 4.29.

Table 4.29 Skenario Use Case Melihat Pembelian Bahan Baku

Melihat Pembelian Bahan Baku	
Actor	Gudang
Objective	Gudang melihat semua daftar pembelian bahan baku
Pre-Condition	Gudang sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Gudang memilih menu Pembelian Bahan Baku 2. Sistem akan menampilkan semua Daftar Pembelian Bahan Baku
Alternative Flow	-
Post-Condition	Gudang berhasil melihat semua daftar pembelian bahan baku

4.2.5.27 Mengubah Pembelian Bahan Baku

Mengubah Pembelian Bahan Baku merupakan fungsi untuk Gudang mengubah data pada daftar pembelian bahan baku dengan yang baru. Skenario diagram Mengubah Pembelian Bahan Baku akan dijelaskan pada table 4.30.

Table 4.30 Skenario Use Case Mengubah Pembelian Bahan Baku

Mengubah Pembelian Bahan Baku	
Actor	Gudang
Objective	Gudang mengubah pembelian bahan baku yang sudah berada pada database konveksi dan sablon
Pre-Condition	<ol style="list-style-type: none"> 1. Gudang sudah berada didalam system 2. Data pembelian sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Gudang memilih Menu Pembelian Bahan Baku 2. Sistem akan menampilkan semua daftar pembelian bahan baku 3. Gudang memilih fungsi tombol Edit bahan baku 4. Sistem menampilkan form pembelian bahan baku yang dipilih berisi tanggal pembelian , nama bahan baku, jumlah bahan, dan catatan 5. Gudang mengubah form pembelian bahan baku sesuai keinginan
Alternative Flow	<ol style="list-style-type: none"> 1. Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”

Post-Condition	Gudang berhasil mengubah pembelian bahan baku menjadi data yang baru ke dalam sistem
----------------	--

4.2.5.28 Menghapus Pesanan

Menghapus Pesanan merupakan fungsi untuk menghapus pesanan yang berada pada daftar pesanan. Skenario diagram menghapus pesanan akan dijelaskan pada table 4.31.

Table 4.31 Skenario Use Case Menghapus Pesanan

Menghapus Pesanan	
Actor	Admin
Objective	Admin menghapus pesanan yang berada pada daftar pesanan
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam sistem 2. Data pesanan sudah berada didalam database
Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu daftar Pesanan 2. Sistem akan menampilkan semua daftar Pesanan yang didalamnya terdapat informasi pesanan 3. Admin memilih fungsi tombol hapus pada salah satu pesanan 4. Sistem menghapus pesanan dari database
Alternative Flow	-
Post-Condition	Admin telah berhasil menghapus pesanan dari database

4.2.5.29 Menghapus Bahan Konveksi dan Sablon

Menghapus Bahan Konveksi dan Sablon merupakan fungsi untuk menghapus pesanan yang berada pada daftar bahan konveksi maupun. Skenario diagram menghapus Menghapus Bahan Konveksi dan Sablon akan dijelaskan pada table 4.32.

Table 4.32 Skenario Use Case Menghapus Bahan Konveksi dan Sablon

Menghapus Bahan Konveksi dan Sablon	
Actor	Admin
Objective	Admin menghapus data bahan baku yang berada pada konveksi maupun
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam sistem 2. Data Bahan Baku sudah berada pada database
Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu Daftar bahan baku 2. Sistem akan menampilkan semua daftar bahan baku yang berada pada konveksi maupun sablon 3. Admin Konveksi memilih fungsi tombol hapus pada salah satu bahan baku

	4. Sistem menghapus bahan baku dari database
Alternative Flow	-
Post-Condition	Admin telah berhasil menghapus bahan baku dari database

4.2.5.30 Menghapus Pembelian Bahan Baku

Menghapus Pembelian Bahan Baku merupakan fungsi untuk menghapus data pembelian bahan baku yang berada didalam database. Skenario diagram Menghapus Pembelian Bahan Baku akan dijelaskan pada table 4.33.

Table 4.33 Skenario Use Case Menghapus Pembelian Bahan baku

Menghapus Pembelian Bahan Baku	
Actor	Admin
Objective	Admin menghapus pembelian bahan baku yang berada didalam database
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam sistem 2. Data pembelian bahan baku sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu Pembelian Bahan Baku 2. Sistem akan menampilkan Semua daftar Pembelian Bahan Baku 3. Admin memilih fungsi tombol hapus pada salah satu Pembelian 4. Sistem menghapus data pembelian bahan baku dari database
Alternative Flow	-
Post-Condition	Admin berhasil menghapus pembelian bahan baku dari database

4.2.5.31 Menghapus Jadwal Induk Produksi

Menghapus Jadwal Induk Produksi merupakan fungsi untuk menghapus laporan jadwal induk produksi yang berada didalam database. Skenario diagram Menghapus Jadwal Induk Produksi akan dijelaskan pada table 4.34.

Table 4.34 Skenario Use Case Menghapus Jadwal Induk Produksi

Menghapus Jadwal Induk Produksi	
Actor	Admin
Objective	Admin menghapus laporan jadwal induk produksi yang berada didalam database
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam sistem 2. Data jadwal induk produksi sudah dimasukkan

Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu Jadwal Produksi 2. Sistem akan menampilkan Semua Laporan yang didalamnya terdapat informasi jadwal induk produksi 3. Admin memilih fungsi tombol hapus pada salah satu Jadwal 4. Sistem menghapus Jadwal Induk Produksi dari database
Alternative Flow	-
Post-Condition	Admin berhasil menghapus Jadwal Produksi dari database

4.2.5.32 Menghapus Request Order

Menghapus Request Order merupakan fungsi untuk menghapus Request Order yang berada didalam database. Skenario diagram Menghapus Request Order akan dijelaskan pada table 4.35.

Table 4.35 Skenario Use Case Menghapus Request Order

Menghapus Request Order	
Actor	Admin
Objective	Adminmenghapus Request Order yang berada didalam database
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam sistem 2. Data Request Order sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu Request Order 2. Sistem akan menampilkan Semua Daftar Request Order 3. Admin memilih fungsi tombol hapus pada salah satu Data Request Order 4. Sistem menghapus Request Order dari database
Alternative Flow	-
Post-Condition	Admin berhasil menghapus Request Order dari database

4.2.5.33 Menambah User

Menambah User merupakan fungsi untuk Menambahkan user baru untuk bisa mengakses sistem. Skenario diagram Menambah User akan dijelaskan pada table 4.36.

Table 4.36 Skenario Use Case Menambah User

Menambah User	
Actor	Admin
Objective	Admin menambahkan user baru untuk bisa mengakses sistem
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam menu Tambah User 2. Sistem telah menampilkan form user
Main Flow	<ol style="list-style-type: none"> 1. Admin mengisi form user yang berisi nama user, username dan password

	2. Admin memilih fungsi selesai untuk melanjutkan ke dalam menu daftar user
Alternative Flow	Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Admin berhasil membuat user

4.2.5.34 Mengubah User

Mengubah User merupakan fungsi untuk Mengubah data user yang sudah ada pada database. Skenario diagram Mengubah User akan dijelaskan pada table 4.37.

Table 4.37 Skenario Use Case Mengubah User

Mengubah User	
Actor	Admin
Objective	Admin mengubah data user yang sudah berada pada database
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam sistem 2. Data user sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu Daftar User 2. Sistem akan menampilkan semua daftar user yang sudah ada pada database 3. Admin memilih fungsi tombol ubah 4. Sistem menampilkan form user yang dipilih berisi nama user, username dan password 5. Admin mengubah form user sesuai keinginan
Alternative Flow	Jika terdapat kolom yang belum di isi maka tidak dapat melanjutkan ke halaman selanjutnya dan akan mengeluarkan tulisan “kolom tidak boleh kosong”
Post-Condition	Admin berhasil mengubah data user dengan yang baru

4.2.5.35 Melihat User

Melihat User merupakan fungsi untuk melihat semua data user pad database. Skenario diagram Melihat User akan dijelaskan pada table 4.38.

Table 4.38 Skenario Use Case Melihat User

Mengubah User	
Actor	Admin
Objective	Admin melihat semua data user pada database
Pre-Condition	Admin sudah berada didalam sistem
Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu Daftar user

	2. Sistem akan menampilkan Semua daftar user yang berada pada database
Alternative Flow	-
Post-Condition	Admin berhasil melihat semua data user

4.2.5.36 Menghapus User

Menghapus User merupakan fungsi untuk menghapus data user yang berada didalam database. Skenario diagram Menghapus user akan dijelaskan pada table 4.39.

Table 4.39 Skenario Use Case Menghapus User

Menghapus User	
Actor	Admin
Objective	Admin menghapus data user yang berada didalam database
Pre-Condition	<ol style="list-style-type: none"> 1. Admin sudah berada didalam sistem 2. Data user sudah dimasukkan
Main Flow	<ol style="list-style-type: none"> 1. Admin memilih menu Daftar User 2. Sistem akan menampilkan Semua data user yang didalamnya terdapat informasi user 3. Admin memilih fungsi tombol hapus pada salah satu user 4. Sistem menghapus data user dari database
Alternative Flow	-
Post-Condition	Admin berhasil menghapus data user dari database

4.3 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) menggambarkan database secara konseptual, ERD sistem manajemen bahan baku dan pemesanan akan digambarkan pada Gambar 4.2. Terdapat 11 entitas diantaranya user, bahan, catatan, catatan bahan, pesanan, jadwal produksi, produksi pesanan, pembelian, pembelian bahan, request order, request order bahan.



BAB 5 PERANCANGAN DAN IMPLEMENTASI

5.1 Perancangan

Proses perancangan dalam pengembangan perangkat lunak berdasarkan model *waterfall* dilakukan ketika proses rekayasa kebutuhan telah selesai dilakukan. Perancangan sistem menggunakan metode OOD (*Object Oriented Design*) dan menggunakan UML (*Unified Modeling language*). Pada sistem manajemen perencanaan bahan baku dan pemesanan pada sablon malang dibagi menjadi empat tahap perancangan yaitu perancangan arsitektur, perancangan Struktur data, perancangan Antar Muka, dan perancangan Komponen.

5.1.1 Perancangan Arsitektur

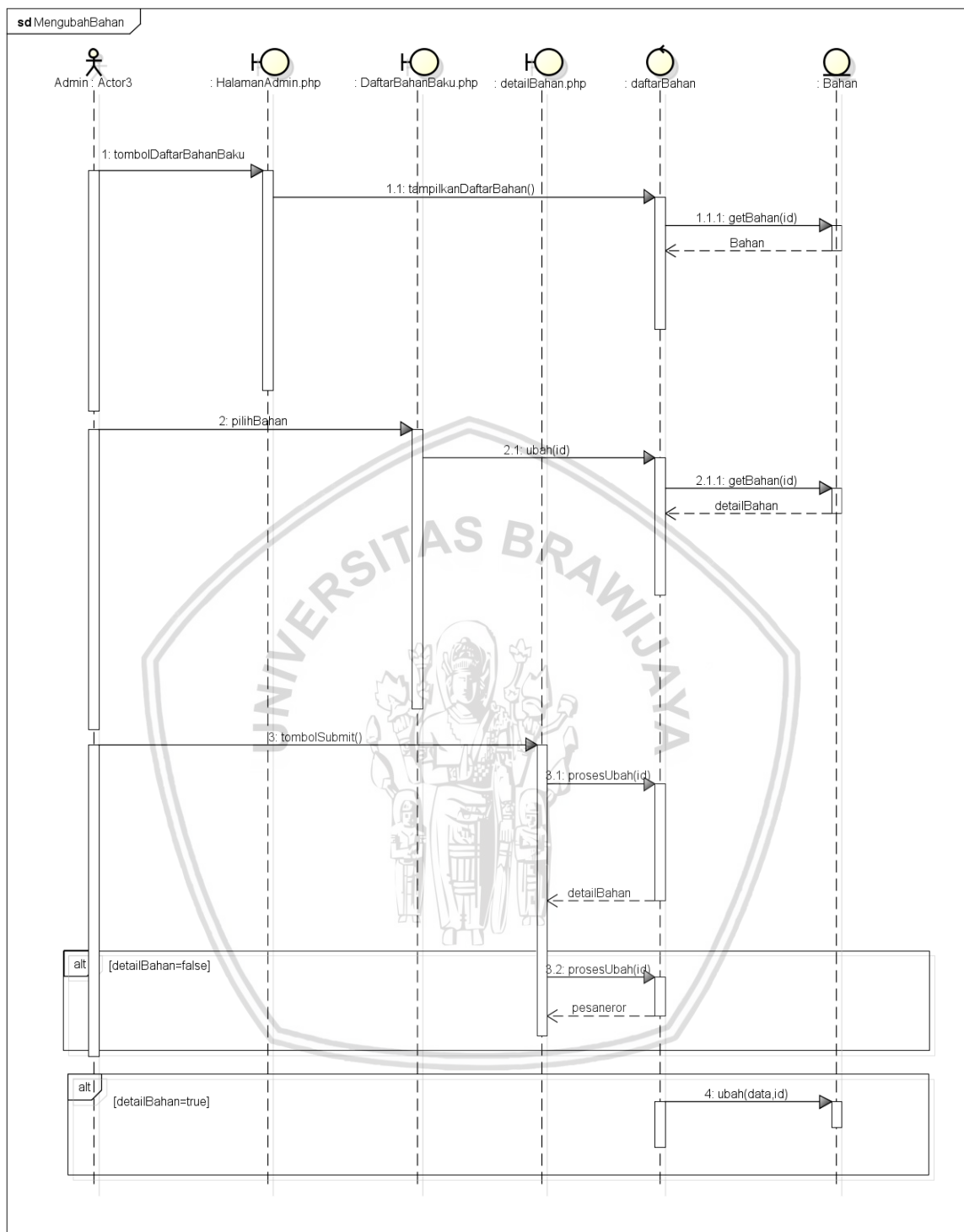
Perancangan arsitektur dari sistem ini akan menggunakan UML (*Unified Modeling language*) yaitu *sequence diagram* dan *class diagram*.

5.1.1.1 Sequence Diagram

Sequence diagram dari sistem manajemen bahan baku dan pemesanan pada sablon malang dibuat berdasarkan *use case* yang didapat dari rekayasa kebutuhan. Sehingga didapat 36 *sequence diagram* yang dibuat. Namun hanya 3 *sequence diagram* yang akan ditampilkan sebagai sampel yaitu *sequence diagram* untuk kebutuhan menambah bahan baku, mengubah bahan baku, update status pemesanan konveksi.

1. *Sequence Diagram* Mengubah Bahan Baku

Sequence Diagram dari Mengubah Bahan Baku yaitu proses untuk mengubah bahan baku yang sudah ada didalam database menjadi bahan baku baru yang sesuai. Proses dimulai dari admin memilih menu daftar bahan baku pada "HalamanAdmin" kemudian mengirim ke controller "daftarBahan" untuk mengambil daftar bahan baku pada database "bahan". Kemudian controller "daftarBahan" akan menampilkan daftar bahan sesuai apa yang sudah dikirimkan kedalam "daftarBahanBaku". Aktor akan memilih salah satu bahan baku yang berada pada daftar bahan baku untuk diubah form bahan baku tersebut. "daftarBahanBaku" meminta kepada controller "daftarBahan" untuk mengubah data dari bahan baku yang dipilih kepada database "bahan". Setelah mendapatkan data bahan maka controller "daftarBahan" akan menampilkan "daftarbahan" sesuai yang dipilih dan aktor akan mengubah form bahan tersebut sesuai keinginan. Terakhir form bahan akan dikirim ke controller "daftarBahan" untuk diproses dan diubah didalam database "bahan". Jika pengisian form pada "Detailbahan" ada yang kosong maka akan mengeluarkan pesan gagal. Gambar *sequence diagram* Mengubah Bahan Baku akan dijelaskan pada gambar 5.1.

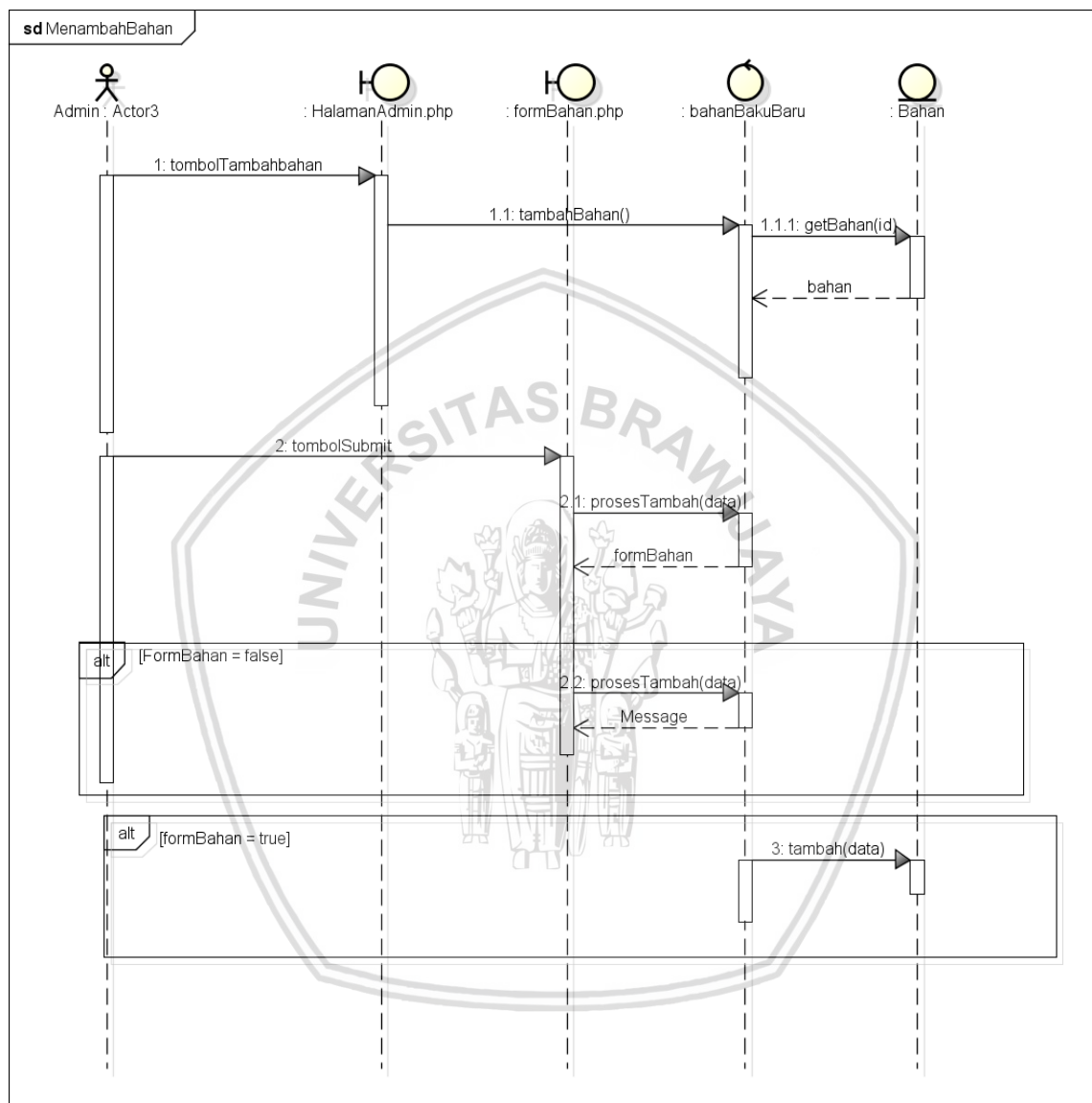


Gambar 5.1 Sequence diagram Mengubah Bahan Baku

2. Sequence Diagram Menambah Bahan Baku

Sequence Diagram dari Menambah Bahan Baku yaitu proses untuk membuat data bahan baku baru ke dalam database. Proses dimulai dari Admin memilih menu tambah bahan pada halaman konveksi kemudian “HalamanAdmin” akan mengirim ke controller “bahanBakuBaru” untuk menampilkan “formBahan”. Aktor mengisi

form bahan baku baru jika sudah selesai maka akan meminta ke controller “bahanBakuBaru” untuk menambahkan data bahan baku tersebut ke dalam database “bahan” untuk diproses. Ketika salah satu form bahan baku ada yang kosong maka akan menampilkan pesan gagal pada “formBahan”. Gambar *sequence diagram* Menambah Bahan Baku akan dijelaskan pada gambar 5.2.

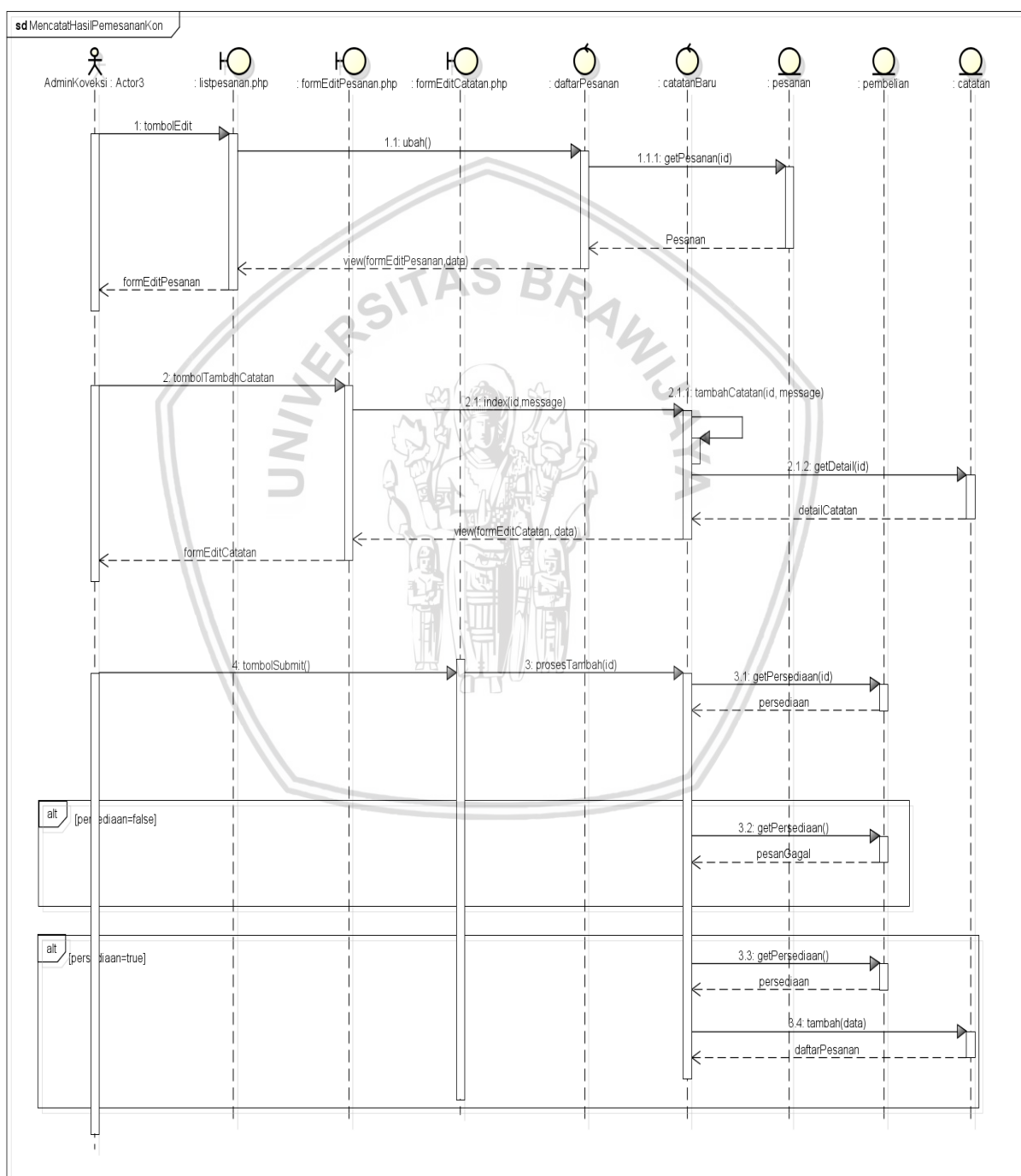


Gambar 5.2 Sequence diagram Menambah Bahan Baku

3. Sequence diagram Mencatat Hasil Pemesanan Konveksi

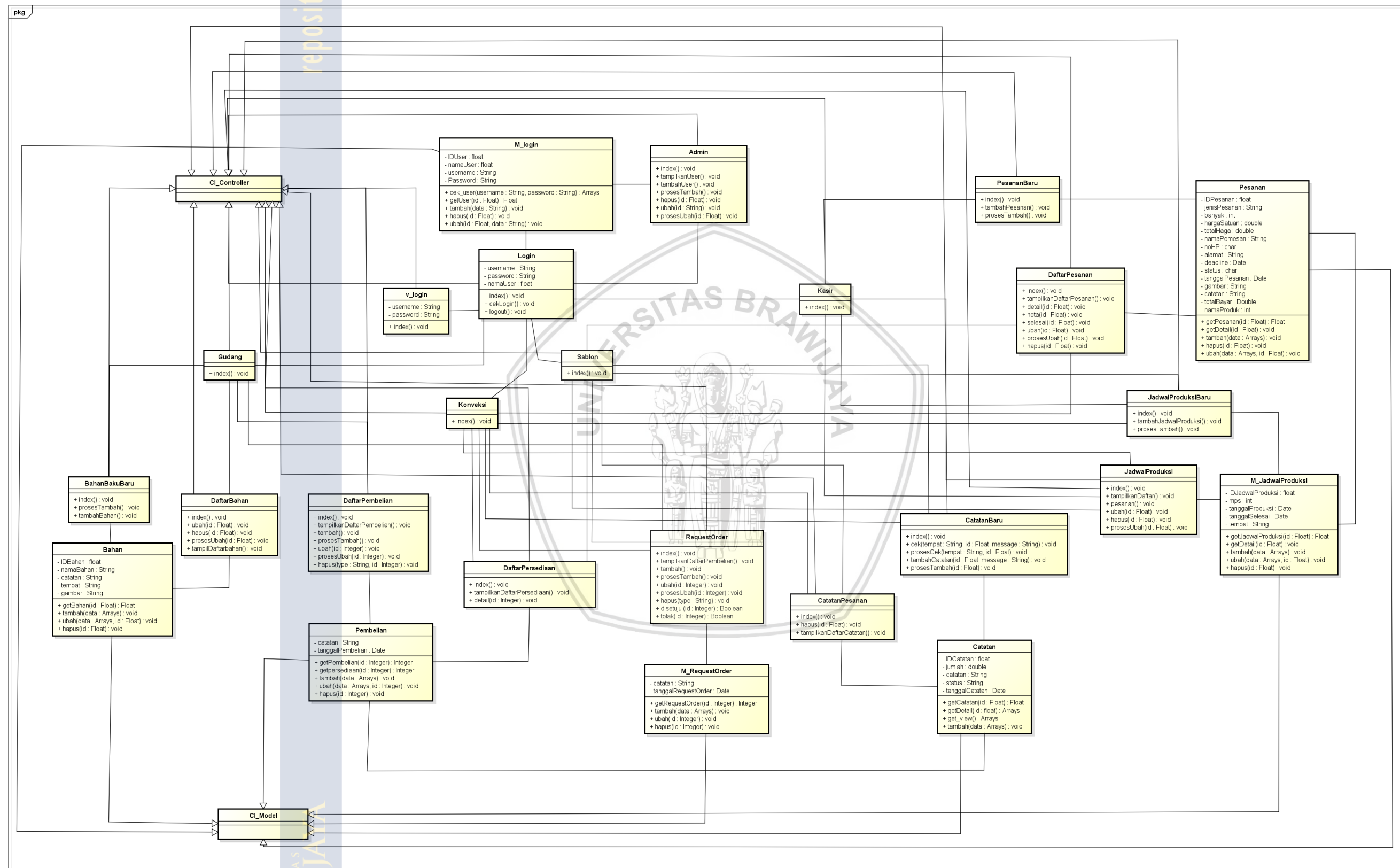
Sequence diagram dari Mencatat Hasil Pemesanan Konveksi yaitu proses untuk mencatat bahan baku yang habis dari pesanan untuk dilanjutkan ke proses selanjutnya. Proses pertama yaitu admin konveksi memilih tombol edit pada “listPesanan” untuk meminta controller “daftarPesanan” mengambil data pesanan pada database “pesanan”. Kemudian data pesanan akan dikirim kembali pada controller “daftarPesanan” yang nantinya akan menampilkan “formEditPesanan”. Aktor akan memilih tombol Tambah Catatan untuk meminta ke controller

“catatanBaru” agar mengambil data catatan yang sudah di isi. Aktor mengisi “formEditCatatan” untuk membuat catatan pada pesanan tersebut. Setelah aktor mengisi “formEditCatatan” maka akan mengirimkan data catatan tersebut kedalam database “catatan” melalui controller “catatanBaru”. Jika pengisian form pada “formEditCatatan” ada yang kosong atau tidak sesuai maka akan mengeluarkan pesan gagal. Gambar *sequence diagram* Mencatat Hasil Pemesanan Konveksi akan dijelaskan pada gambar 5.3.



Gambar 5.3 Sequence diagram Mencatat Hasil Pesanan Konveks

5.1.1.2 Class Diagram



Gambar 5.4 Class Diagram

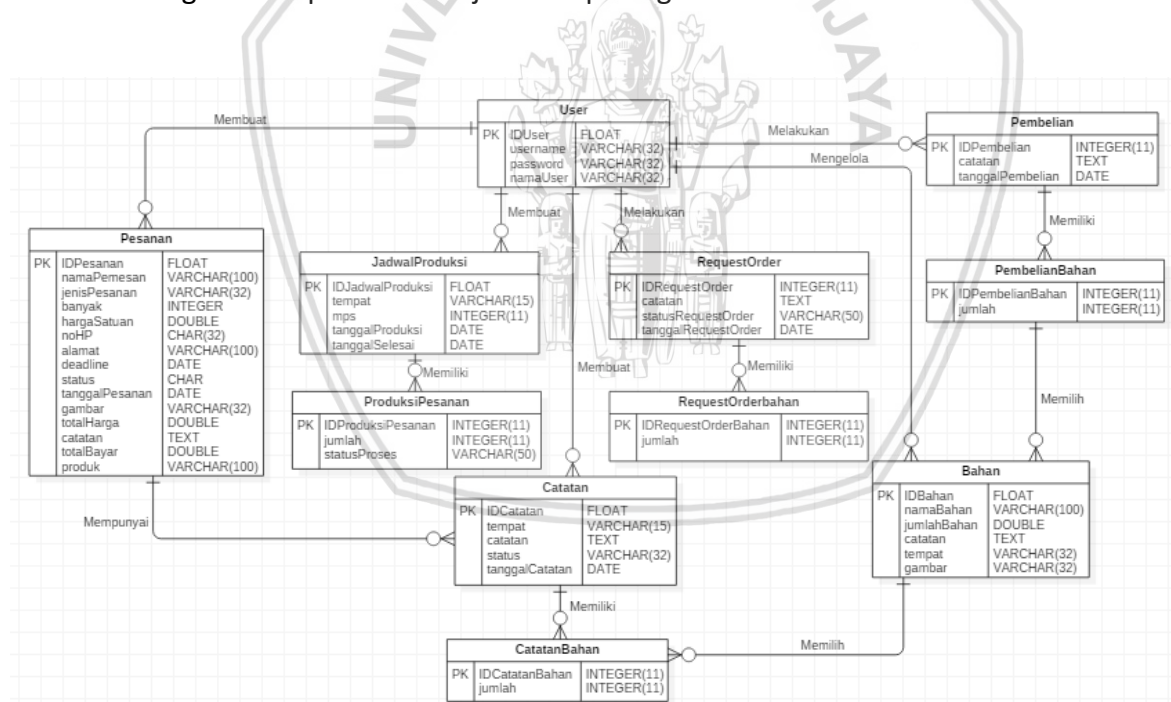
Gambar 5.4 merupakan *clas diagram* yang digunakan pada pengembangan sistem aplikasi. Pada diagram ini terdapat kelas *controller* yang membatasi antara user dan sistem serta beberapa model yang bertugas mengakses *database*. Kelas *controller* yang terdapat pada sistem ini antara lain "CI_Controller", "Admin", "BahanBakuBaru", "CatatanBaru", "CatatanPesanan", "DaftarBahan", "DaftarPembelian", "DaftarPersediaan", "DaftarPesanan", "JadwalProduksi", "JadwalProduksiBaru", "Kasir", "Konveksi", "Login", "Gudang", "PesananBaru", "RequestBaru", "RequestOrder", dan "Sablon". Sedangkan untuk *entity* yaitu "CI_Model", "Bahan", "Catatan", "Pesanan", "M_login", "M_jadwalProduksi", "M_requestOrder", dan "Pembelian".

5.1.2 Perancangan Struktur Data

Perancangan struktur data terdapat rancangan tabel basis data yang akan digunakan oleh sistem ini adalah dalam bentuk ERD (*Entity Relationship Diagram*) dan PDM (*Physical Data Model*).

5.1.2.1 Conceptual Data Model (CDM)

CDM adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas serta hubungan antara entitas-entitas itu. Perancangan CDM pada akan dijelaskan pada gambar 5.5.



Gambar 5.5 Conceptual Data Model

5.1.2.2 Physical Data Model (PDM)

Physical Data Model (PDM) dilakukan berdasarkan apa yang sudah dibuat pada perancangan *Entity Relationship Diagram* (ERD) sebelumnya. Jumlah tabel yang akan dibuat sama dengan jumlah entitas yang telah didefinisikan pada ERD. Terdapat beberapa tabel dalam database yang di identifikasikan yaitu tabel user yang berisi database setiap

```

    erDiagram
        User ||--o{ Pesanan : "PK IDUser"
        User ||--o{ Pembelian : "PK IDPembelian"
        JadwalProduksi ||--o{ Pesanan : "PK IDJadwalProduksi"
        RequestOrder ||--o{ Pesanan : "PK IDRequestOrder"
        RequestOrder ||--o{ RequestOrderBahan : "PK IDRequestOrderBahan"
        Pembelian ||--o{ PembelianBahan : "PK IDPembelianBahan"
        Catatan ||--o{ Pesanan : "FK IDPesanan"
        Catatan ||--o{ CatatanBahan : "FK IDCatatanBahan"
        Bahan ||--o{ CatatanBahan : "FK IDBahan"

        User {
            string username PK
            string password
            string namaUser
        }

        Pesanan {
            string IDPesanan PK
            string namaPemesan
            string jenisPesanan
            float banyak
            float hargaSatuan
            string noHP
            string alamat
            string deadline
            string status
            string tanggalPesanan
            string gambar
            float totalHarga
            string catatan
            float totalBayar
            string produk
        }

        JadwalProduksi {
            string IDJadwalProduksi PK
            float tempat
            string mps
            string tanggalProduksi
            string tanggalSelesai
        }

        RequestOrder {
            string IDRequestOrder PK
            string catatan
            string statusRequestOrder
            string tanggalRequestOrder
        }

        RequestOrderBahan {
            string IDRequestOrderBahan PK
            string IDRequestOrder FK
            string IDBahan FK
            float jumlah
        }

        Pembelian {
            string IDPembelian PK
            string catatan
            string tanggalPembelian
        }

        PembelianBahan {
            string IDPembelianBahan PK
            string IDPembelian FK
            string IDBahan FK
            float jumlah
        }

        Catatan {
            string IDCatatan PK
            string IDPesanan FK
            float tempat
            string catatan
            string status
            string tanggalCatatan
        }

        CatatanBahan {
            string IDCatatanBahan PK
            string IDCatatan FK
            string IDBahan FK
            float jumlah
        }

        Bahan {
            string IDBahan PK
            string namaBahan
            float jumlahBahan
            float DOUBLE
            string catatan
            string tempat
            string gambar
        }

```

5.1.3 Perancangan Komponen

Nama Klas : CatatanBaru
Nama Operasi : prosescek
Algoritma :

NO	Pseudocode
----	------------

1	<i>If (post=inputan)</i>
2	<i>Instansiasi variable pesan = false</i>
3	<i>Mengambil data persediaan</i>
4	<i>Instansiasi variable array</i>
5	<i>Foreach (data persediaan berdasarkan baris)</i>
6	<i>Data persediaan disimpan dalam variable array</i>
7	<i>Instansiasi variable array</i>
8	<i>Foreach (jumlah inputan bahan)</i>
9	<i>If (data bahan berhasil diinputkan)</i>
10	<i>Inisialisasi data array dengan data inputan</i>
11	<i>End if</i>
12	<i>Else</i>
13	<i>Memasukkan langsung data cek bahan</i>
14	<i>End foreach</i>
15	<i>Foreach (data cek bahan berdasarkan jumlah inputan)</i>
16	<i>If (data bahan berhasil di inputkan)</i>
17	<i>If (jumlah persediaan < inputan data bahan)</i>
18	<i>Menampilkan pesan “Jumlah Melebihi Kapasitas”</i>
19	<i>End if</i>
20	<i>End if</i>
21	<i>Else</i>
22	<i>Menampilkan Pesan “Data Bahan Tidak ditemukan”</i>
23	<i>End foreach</i>
24	<i>Jika (tidak menampilkan pesan error)</i>
25	<i>Inisialisasi variable IDPesanan</i>
26	<i>Inisialisasi variable status</i>
27	<i>Inisialisasi variable tempat</i>
28	<i>Membuat variable untuk memanggil method tambah di database</i>
29	<i>Membuat variable untuk memanggil method getDetail di database</i>
30	<i>Instansiasi variable array</i>
31	<i>Foreach (hasil dari data inputan)</i>
32	<i>If (inputan tempat = data tempat)</i>

33	<i>Inisialisasi variable array dengan data IDProduksiPesanan</i>
34	<i>End if</i>
35	<i>End foreach</i>
36	<i>Update produksi pesanan statuspesanan menjadi proses</i>
37	<i>Instansiasi variable array</i>
38	<i>Foreach (jumlah inputan bahan)</i>
39	<i>If (data bahan berhasil di inputkan)</i>
40	<i>Inisialisasi data array dengan data inputan</i>
41	<i>End if</i>
42	<i>Else</i>
43	<i>Memasukkan langsung data inputan</i>
44	<i>End foreach</i>
45	<i>Memasukkan data catatan dengan data inputan</i>
46	<i>Kembali ke halaman detail pesanan</i>
47	<i>End if</i>
48	<i>Else</i>
49	<i>Memanggil method cek</i>
50	<i>End if</i>

Nama Klas : RequestOrder
 Nama Operasi : prosesTambah
 Algoritma :

NO	Pseudocode
1	<i>If (post=inputan)</i>
2	<i>Inisialisasi variable id ke model requestorder</i>
3	<i>membuat variable array</i>
4	<i>For (Berhenti ketika i < jumlah IDBahan)</i>
5	<i>Inisialisasi variable array berdasarkan</i>
6	<i>IDRequestOrder</i>
7	<i>IDBahan</i>
8	<i>Jumlah</i>
9	<i>End for</i>
10	

11	<i>Menyimpan masukan ke dalam database requestorderbahan dengan variable array</i>
12	
13	<i>Kembali ke requestOrder</i>
14	<i>End if</i>
	<i>Memanggil method tambah</i>

Nama Klas : JadwalProduksiBaru

Nama Operasi : prosesTambah

Algoritma :

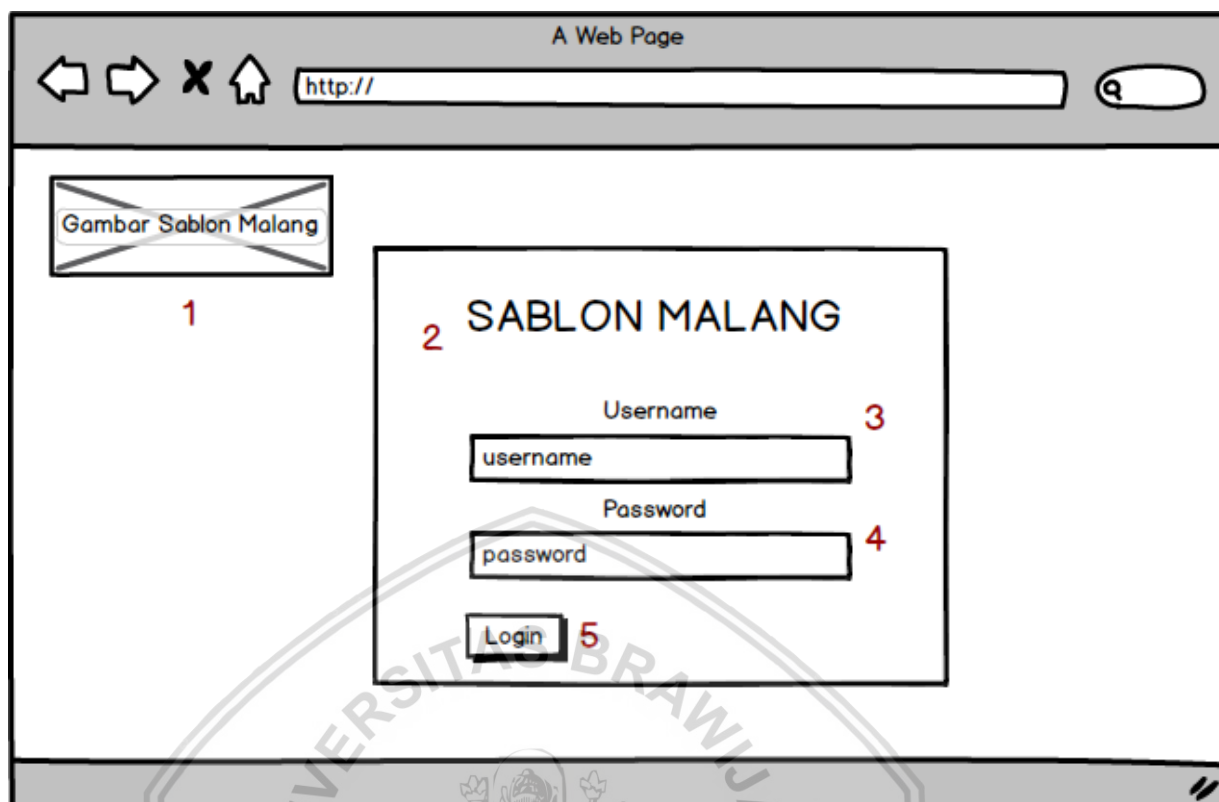
NO	Pseudocode
1	<i>If (post=inputan)</i>
2	<i>If (akses user = konveksi)</i>
3	<i>Memasukkan data jadwal produksi berdasarkan tempat konveksi</i>
4	<i>End if</i>
5	<i>else</i>
6	<i>Memasukkan data jadwal produksi berdasarkan tempat sablon</i>
7	<i>End else</i>
8	<i>Memanggil method tambah dari database jadwal produksi berdasarkan inputan</i>
9	<i>Kembali ke halaman jadwal produksi</i>
10	<i>End if</i>
11	<i>Kembali ke halaman jadwal produksi</i>

5.1.4 Perancangan Antarmuka

Perancangan antar muka terdapat rancangan antar muka dari sistem dalam bentuk tata muka komponen yang ada pada sistem.

1. Halaman Login

Halaman Login adalah halaman utama yang diakses oleh user sebelum masuk kedalam sistem. Halaman Login akan dijelaskan pada gambar 5.7 dan penjelasan detail akan dijelaskan pada table 5.1.



Gambar 5.7 Tampilan Antarmuka Halaman Login

Table 5.1 Penjelasan Antarmuka Halaman Login

No	Nama Object	Keterangan
1	Gambar	Gambar atau logo Sablon Malang
2	Nama Produk	Label yang berisikan tulisan Sablon Malang
3	Username	Text yang nanti akan diisikan user berupa username
4	Password	Text yang nanti akan diisikan user berupa password
4	Login	Tombol untuk masuk kedalam sistem setelah mengisikan username dan password

2. Halaman Tambah Pesanan

Halaman Tambah Pesanan adalah halaman yang berisi form untuk memasukan detail pesanan dari pelanggan. Halaman Tambah Pesanan akan dijelaskan pada gambar 5.8 dan penjelasan detail akan dijelaskan pada table 5.2.

Gambar 5.8 Tampilan Antarmuka Halaman Tambah Pesanan

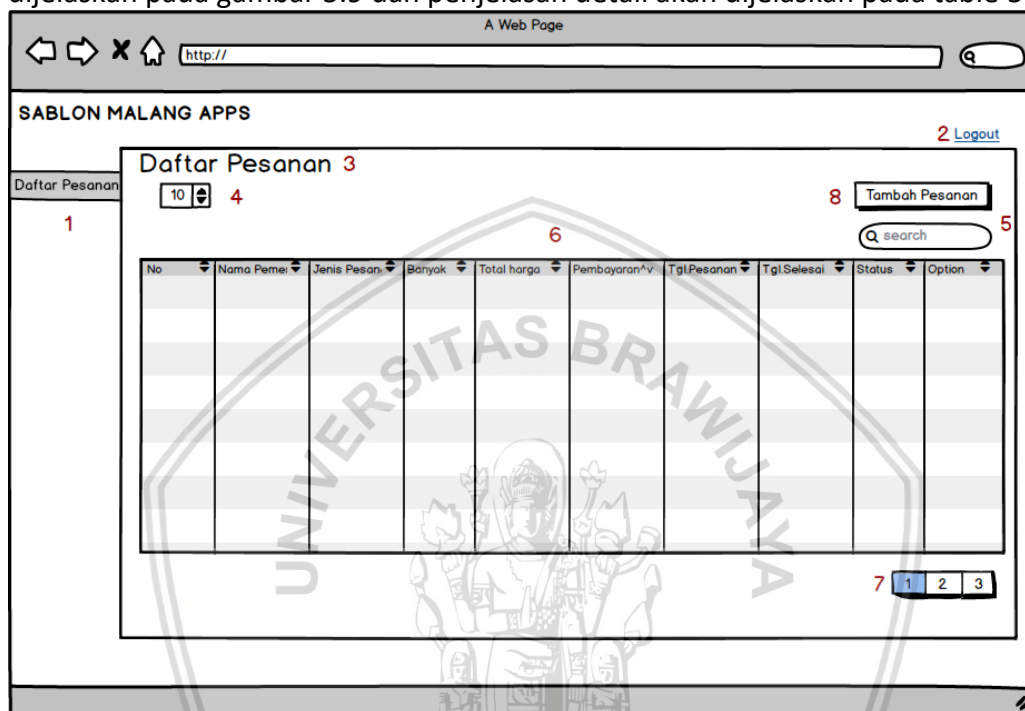
Table 5.2 Penjelasan Antarmuka Halaman Tambah Pesanan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Tambah Pesanan
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Pesanan Baru
4	Form Pesanan	Menampilkan form pesanan yang nanti akan diisi oleh kasir
5	Tombol Submit	Tombol untuk menyelesaikan proses membuat pesanan yang nanti akan disimpan ke dalam database
6	Judul Form	Menampilkan Judul Form yaitu jadwal produksi
7	Tombol Tambah Jadwal	Tombol untuk menampilkan form masukan jadwal produksi baru ke form pesanan

8	Tombol Hapus Jadwal	Tombol untuk menghapus form jadwal dari form pesanan
---	---------------------	--

3. Halaman Daftar Pesanan

Halaman Daftar Pesanan adalah halaman yang berisi daftar pesanan yang dikerjakan dan detail dari pesanan tersebut. Halaman Daftar Pesanan akan dijelaskan pada gambar 5.9 dan penjelasan detail akan dijelaskan pada table 5.3.



Gambar 5.9 Tampilan Antarmuka Halaman Daftar Pesanan

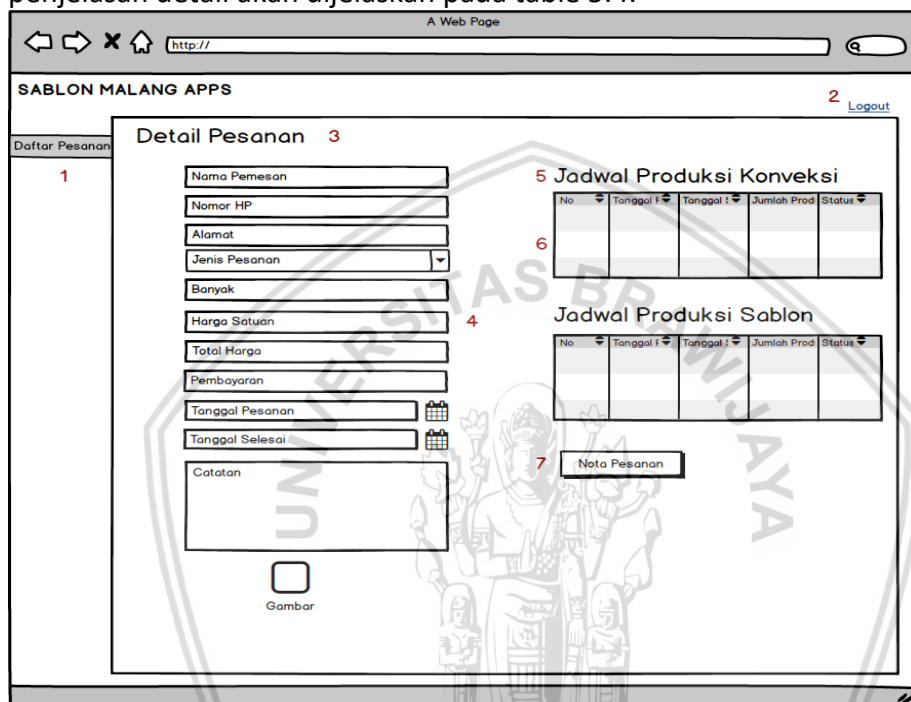
Table 5.3 Penjelasan Antarmuka Halaman Daftar Pesanan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Pesanan
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Pesanan
4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar pesanan yang tersedia
6	Tabel Daftar Pesanan	Menampilkan semua Daftar Pesanan yang dikerjakan baik yang sudah selesai maupun belum dan berisi opsi untuk mengelola pesanan tersebut

7	Button Angka	Menampilkan halaman dari daftar pesanan
8	Button Tambah Pesanan	Tombol untuk menambah pesanan baru dengan menampilkan form pesanan baru

4. Halaman Detail Pesanan

Halaman Detail Pesanan adalah halaman yang berisi detail dari Pesanan dan tidak bisa diubah. Halaman Detail Pesanan akan dijelaskan pada gambar 5.10 dan penjelasan detail akan dijelaskan pada table 5.4.



Gambar 5.10 Tampilan Antarmuka Halaman Ubah Pesanan

Table 5.4 Penjelasan Antarmuka Halaman Ubah Pesanan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Pesanan
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Detail Pesanan
4	Form detail Bahan baku	Menampilkan form detail Pesanan yang dapat diubah
5	Judul Form	Menampilkan Judul dari form
6	Tabel Jadwal Produksi	Menampilkan jadwal produksi dari pesanan
7	Tombol Nota Pesanan	Tombol untuk mencetak Nota Pesanan

5. Halaman Ubah Pesanan

Halaman Ubah Pesanan adalah halaman yang berisi detail dari Pesanan dan bisa diubah sesuai keinginan. Halaman Ubah Pesanan akan dijelaskan pada gambar 5.11 dan penjelasan detail akan dijelaskan pada table 5.5.

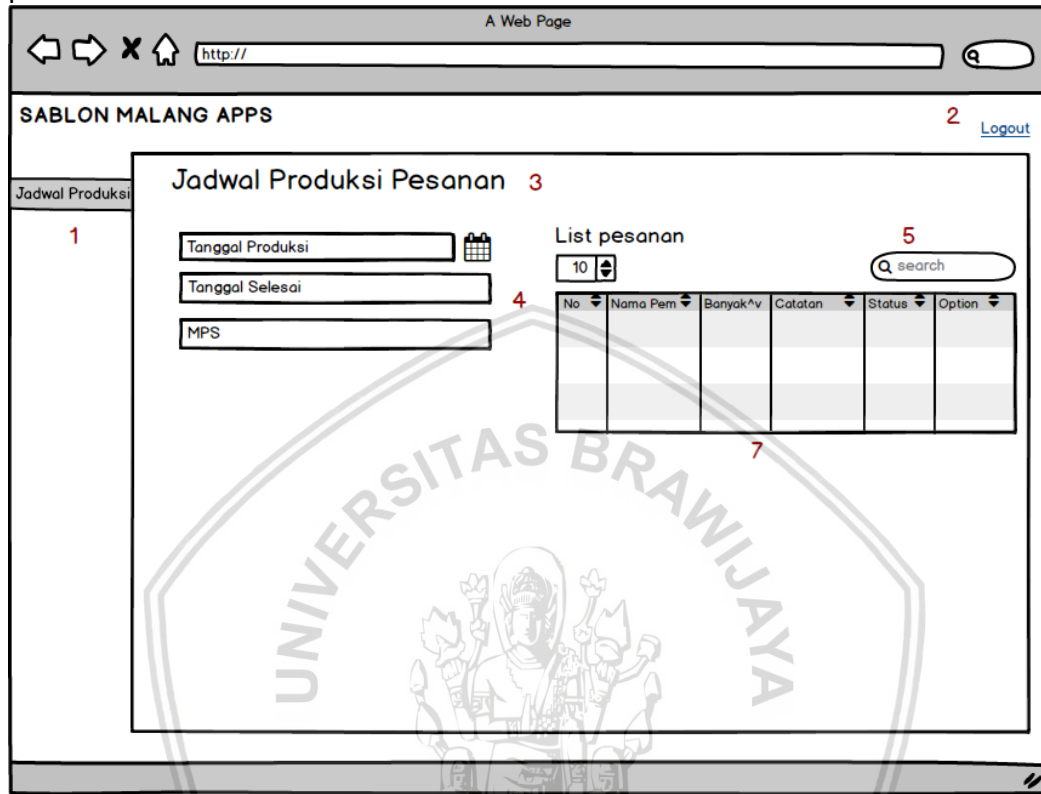
Gambar 5.11 Tampilan Antarmuka Halaman Ubah Pesanan

Table 5.5 Penjelasan Antarmuka Halaman Ubah Pesanan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Pesanan
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Ubah Pesanan
4	Form detail Bahan baku	Menampilkan form detail Pesanan yang dapat diubah
5	Judul Form	Menampilkan Judul dari form
6	Tabel Jadwal Produksi	Menampilkan jadwal produksi dari pesanan
7	Tombol Submit	Tombol yang berfungsi untuk menyimpan form detail bahan baku yang sudah dirubah

6. Halaman Jadwal Produksi Pesanan

Halaman Jadwal Produksi Pesanan adalah halaman yang berisi daftar jadwal produksi beserta pesanan pada jadwal produksi tersebut. Halaman Jadwal Produksi Pesanan akan dijelaskan pada gambar 5.12 dan penjelasan detail akan dijelaskan pada table 5.6.



Gambar 5.12 Tampilan Antarmuka Halaman Jadwal Produksi Pesanan

Table 5.6 Penjelasan Antarmuka Halaman Jadwal Produksi Pesanan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Pesanan
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Jadwal Produksi Pesanan
4	Form Jadwal Produksi	Menampilkan form detail Jadwal Produksi
5	Menu Search	Menu yang berfungsi untuk mencari daftar pesanan yang tersedia
6	Judul Form	Menampilkan Judul dari form
7	Tabel List Pesanan	Menampilkan daftar pesanan yang terdapat pada jadwal produksi tersebut dengan option untuk mengelola pesanan tersebut

7. Halaman Check Persediaan

Halaman Check Persediaan adalah halaman yang berisi inputan bahan baku yang akan dicek. Halaman Check Persediaan akan dijelaskan pada gambar 5.13 dan penjelasan detail akan dijelaskan pada table 5.7.

Gambar 5.13 Tampilan Antarmuka Halaman Check Persediaan

Table 5.7 Penjelasan Antarmuka Halaman Check Persediaan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Jadwal Produksi
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Catatan Pesanan
4	Form Catatan Pesanan	Menampilkan form Catatan Pesanan
5	Tombol Submit	Tombol untuk menyimpan dan mengecek data bahan baku
6	Tombol Tambah Bahan	Tombol untuk menambahkan form bahan
7	Form Detail Pesanan	Menampilkan detail pesanan yang dipilih

8. Halaman Tambah Catatan

Halaman Tambah Catatan adalah halaman yang berisi form catatan pesanan yang sudah selesai dikerjakan. Halaman Tambah Catatan akan dijelaskan pada gambar 5.14 dan penjelasan detail akan dijelaskan pada table 5.8.

Gambar 5.14 Tampilan Antarmuka Halaman Tambah Catatan

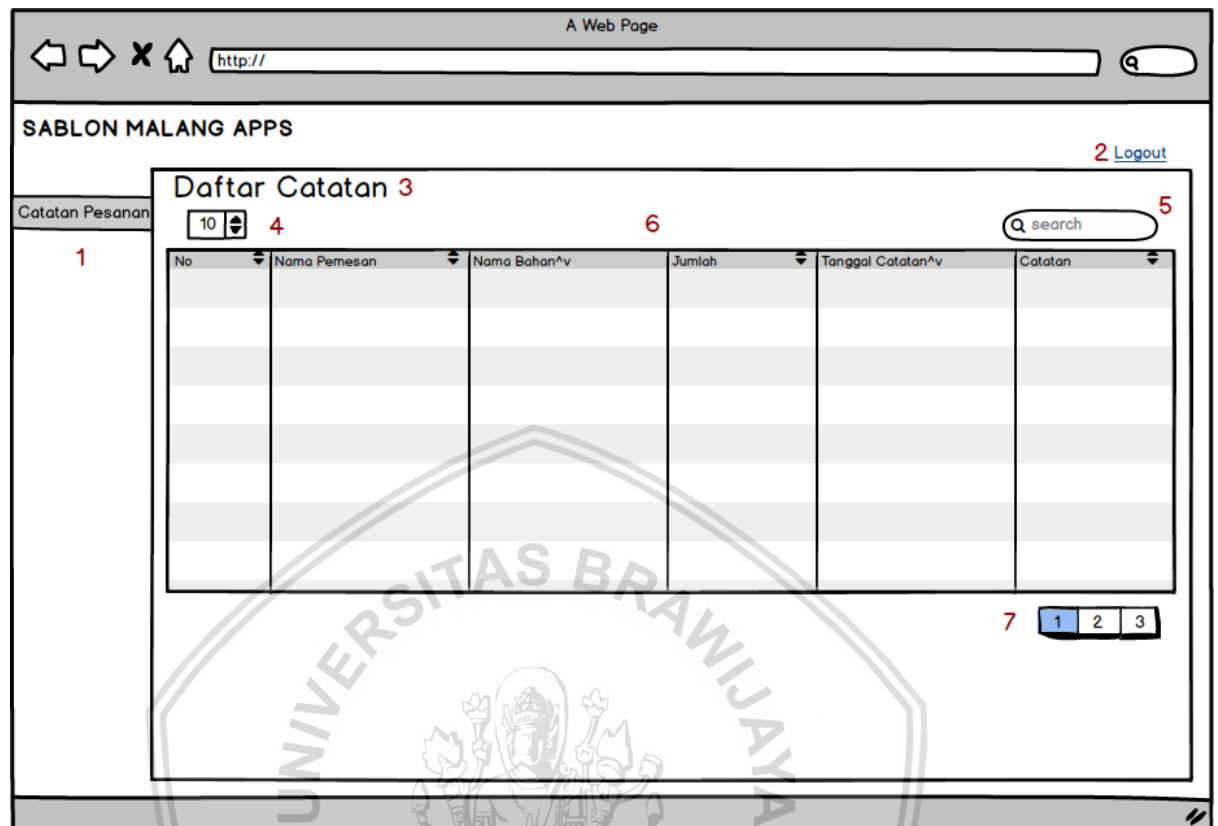
Table 5.8 Penjelasan Antarmuka Halaman Tambah Catatan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Jadwal Produksi
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Catatan Pesanan
4	Form Catatan Pesanan	Menampilkan form Catatan Pesanan
5	Tombol Submit	Tombol untuk menyimpan catatan pesanan
6	Form Detail Pesanan	Menampilkan detail pesanan yang dipilih

9. Halaman Daftar Catatan Pesanan

Halaman Daftar Catatan Pesanan adalah halaman yang berisi daftar Catatan Pesanan yang sudah terdapat kedalam database. Halaman Daftar Catatan Pesanan

akan dijelaskan pada gambar 5.15 dan penjelasan detail akan dijelaskan pada table 5.9.



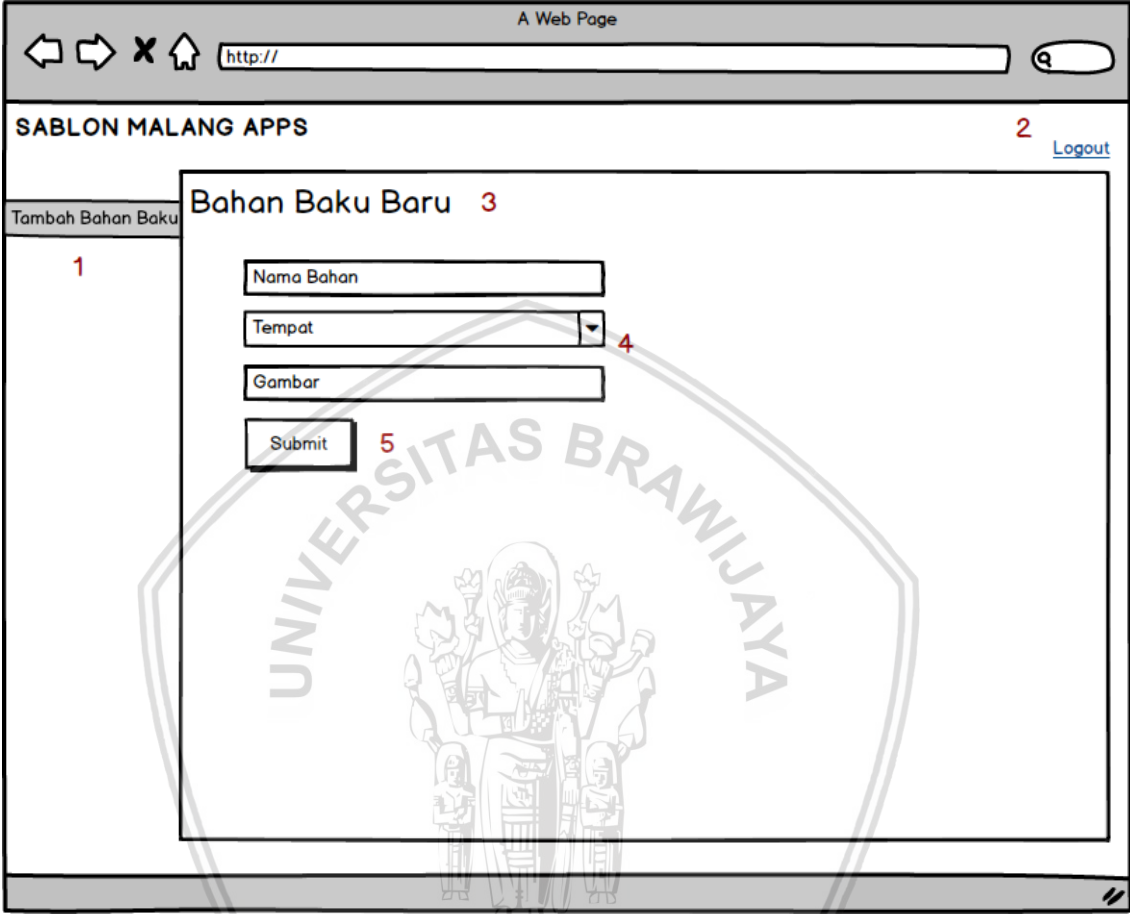
Gambar 5.15 Tampilan Antarmuka Halaman Daftar Catatan Pesanan

Table 5.9 Penjelasan Antarmuka Halaman Daftar Catatan Pesanan

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Catatan Pesanan
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Catatan
4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar bahan baku yang tersedia
6	Tabel Catatan Pesanan	Menampilkan semua Catatan Pesanan
7	Button Angka	Menampilkan halaman dari daftar catatan pesanan

10. Halaman Tambah Bahan Baku

Halaman Tambah Bahan Baku adalah halaman yang berisi form untuk menambahkan bahan baku baru baik itu konveksi maupun sablon. Halaman Tambah Bahan Baku akan dijelaskan pada gambar 5.16 dan penjelasan detail akan dijelaskan pada table 5.10.



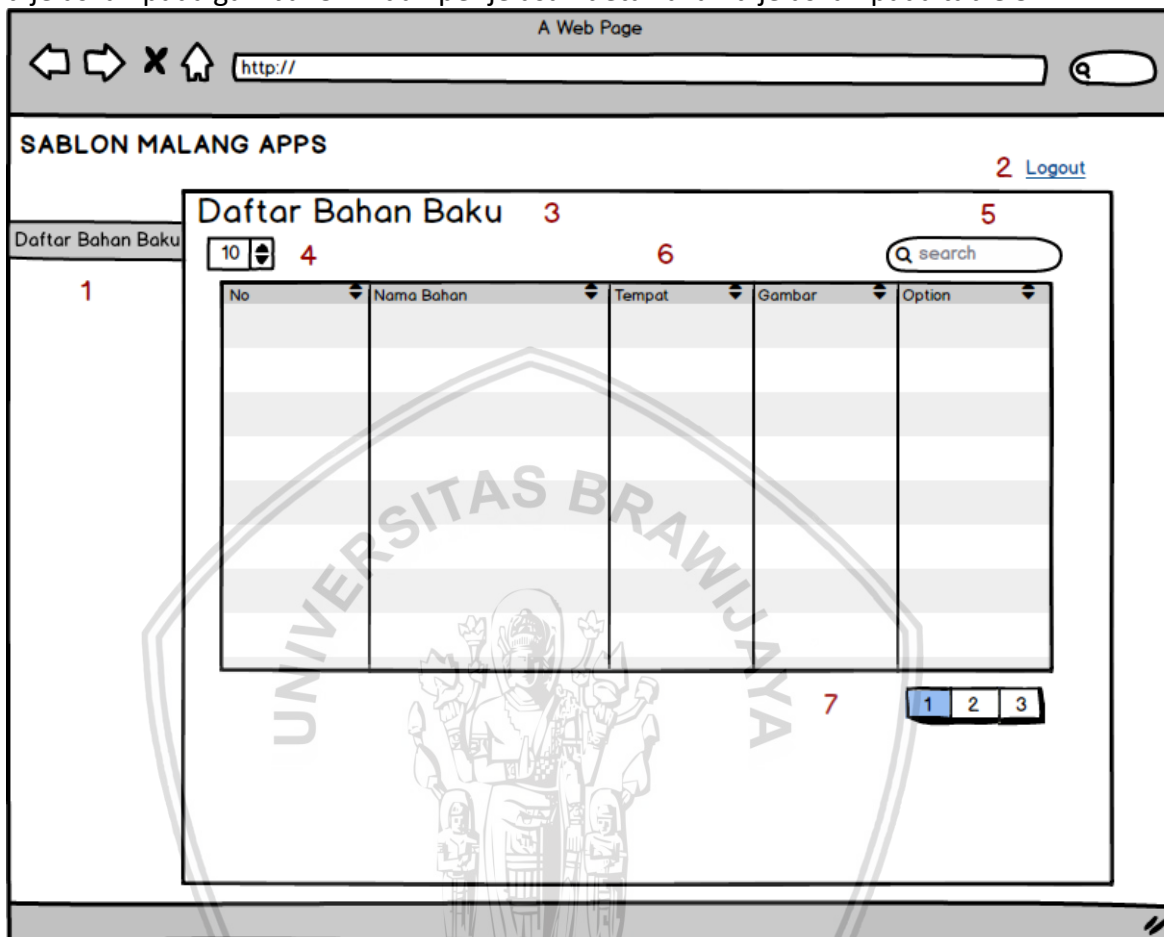
Gambar 5.16 Tampilan Antarmuka Halaman Tambah Bahan Baku

Table 5.10 Penjelasan Antarmuka Halaman Tambah Bahan Baku

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Tambah Bahan Baku
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Bahan Baku Baru
4	Form Bahan Baku Baru	Menampilkan form Tambah Bahan Baku yang akan diisi secara detail
5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form tambah bahan baku yang nanti akan ditambahkan kedalam

11. Halaman Daftar Bahan Baku

Halaman Daftar Bahan Baku adalah halaman yang berisi daftar bahan baku yang sudah terdaftar baik itu konveksi maupun sablon. Halaman Daftar Bahan Baku akan dijelaskan pada gambar 5.17 dan penjelasan detail akan dijelaskan pada table 5.11.



Gambar 5.17 Tampilan Antarmuka Halaman Daftar Bahan Baku

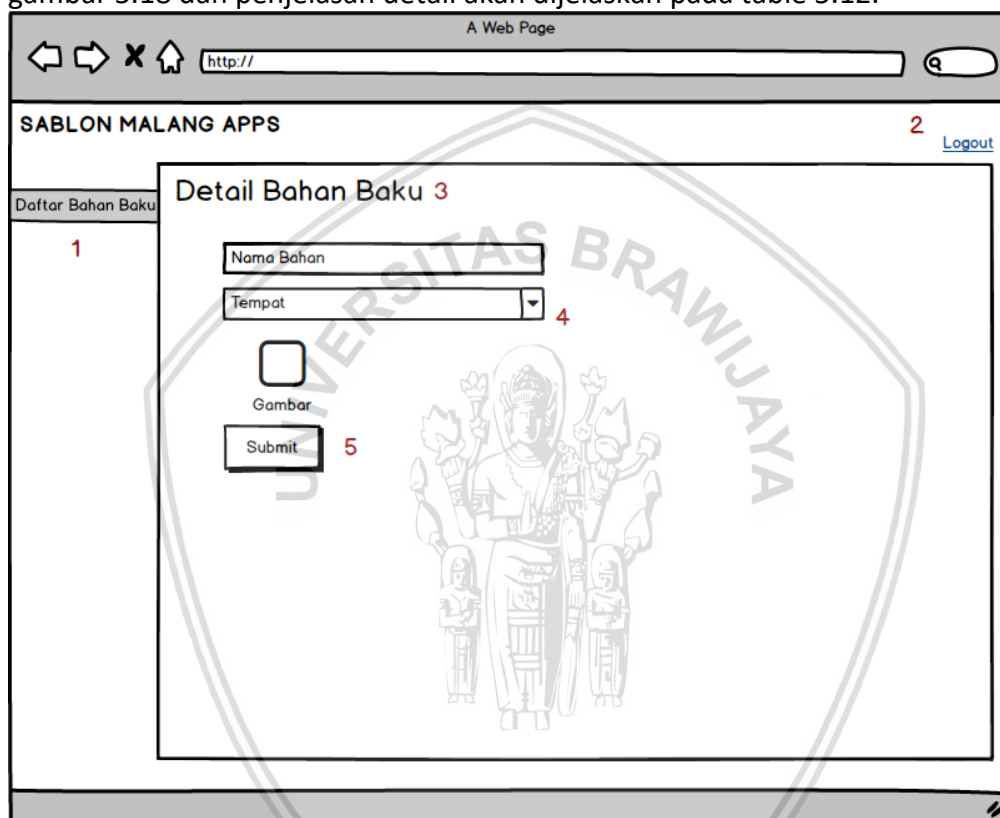
Table 5.11 Penjelasan Antarmuka Halaman Daftar Bahan Baku

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Bahan Baku
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Bahan Baku
4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar bahan baku yang tersedia

6	Tabel Daftar Bahan Baku	Menampilkan semua Daftar Bahan Baku sesuai dengan namaUser dan berisi opsi untuk mengelola Bahan Baku tersebut
7	Button Angka	Menampilkan halaman dari daftar Bahan Baku

12. Halaman Detail Bahan Baku

Halaman Detail Bahan Baku adalah halaman yang berisi detail dari bahan baku dan bisa diubah sesuai keinginan. Halaman Detail Bahan Baku akan dijelaskan pada gambar 5.18 dan penjelasan detail akan dijelaskan pada table 5.12.



Gambar 5.18 Tampilan Antarmuka Halaman Detail Bahan Baku

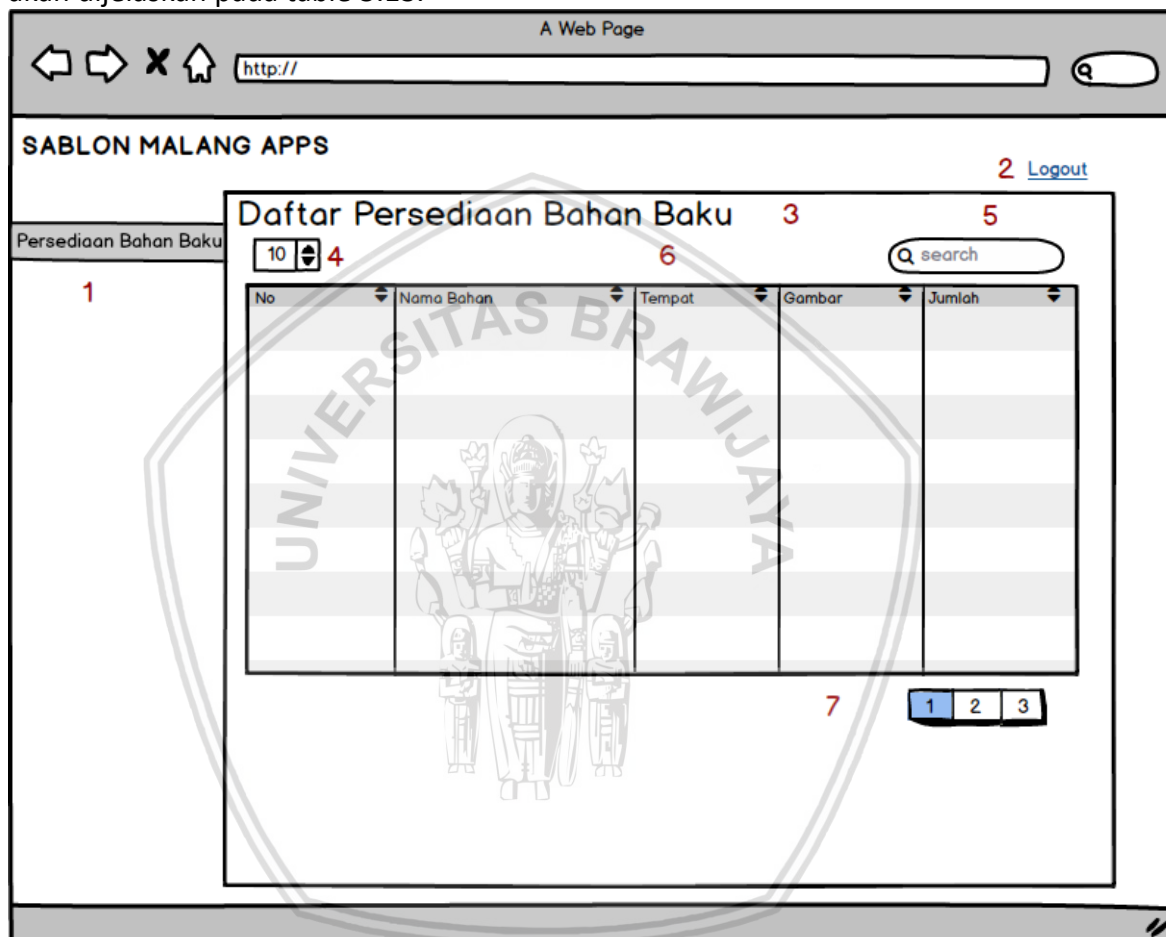
Table 5.12 Penjelasan Antarmuka Halaman Detail Bahan Baku

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Bahan Baku
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Detail Bahan Baku
4	Form detail Bahan baku	Menampilkan form detail bahan baku yang dapat diubah

5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form detail bahan baku yang sudah dirubah
---	---------------	---

13. Halaman Daftar Persediaan Bahan Baku

Halaman Daftar Persediaan Bahan Baku adalah halaman yang berisi daftar Persediaan bahan baku yang sudah terdaftar sesuai hak akses user. Halaman Daftar Persediaan Bahan Baku akan dijelaskan pada gambar 5.19 dan penjelasan detail akan dijelaskan pada table 5.13.



Gambar 5.19 Tampilan Antarmuka Halaman Daftar Persediaan Bahan Baku

Table 5.13 Penjelasan Antarmuka Halaman Daftar Persediaan Bahan Baku

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Persediaan Bahan Baku
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Persediaan Bahan Baku

4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar bahan baku yang tersedia
6	Tabel Daftar Bahan Baku	Menampilkan semua Persediaan Bahan Baku sesuai dengan hak akses user
7	Button Angka	Menampilkan halaman dari Persediaan Bahan Baku

14. Halaman Tambah Request Order

Halaman Tambah Request Order adalah halaman yang berisi form untuk menambahkan Request Order baru baik itu konveksi maupun sablon. Halaman Tambah Request Order akan dijelaskan pada gambar 5.20 dan penjelasan detail akan dijelaskan pada table 5.14.

Gambar 5.20 Tampilan Antarmuka Halaman Tambah Request Order

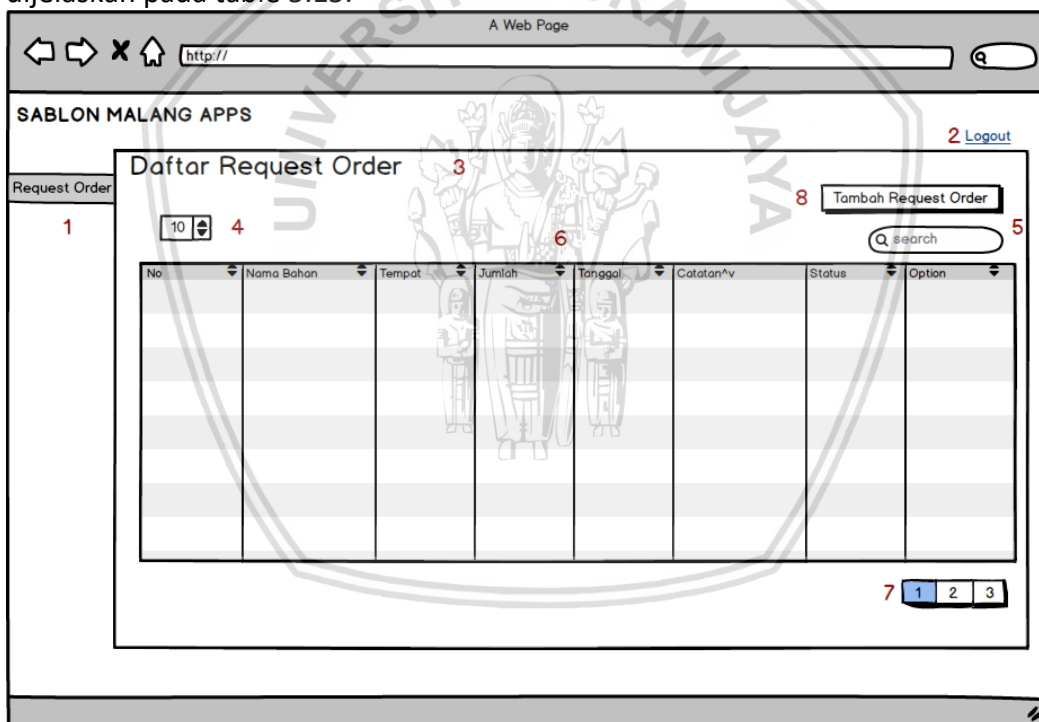
Table 5.14 Penjelasan Antarmuka Halaman Tambah Request Order

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Tambah Request Order
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem

3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Tambah Request Order
4	Form Request Order	Menampilkan form Tambah Request Order yang akan diisi secara detail
5	Tombol Tambah	Tombol yang digunakan untuk menambah form Request Order
6	Tombol Submit	Tombol yang berfungsi untuk menyimpan form tambah Request Order yang nanti akan ditambahkan kedalam

15. Halaman Daftar Request Order

Halaman Daftar Request Order adalah halaman yang berisi daftar Request Order yang sudah terdaftar baik itu konveksi maupun sablon sesuai hak akses. Halaman Daftar Request Order akan dijelaskan pada gambar 5.21 dan penjelasan detail akan dijelaskan pada table 5.15.



Gambar 5.21 Tampilan Antarmuka Halaman Daftar Request Order

Table 5.15 Penjelasan Antarmuka Halaman Daftar Request Order

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Request Order
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem

3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Request Order
4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar bahan baku yang tersedia
6	Tabel Daftar Request Order	Menampilkan semua Daftar Request Order sesuai dengan namaUser dan berisi opsi untuk mengelola Request Order tersebut
7	Button Angka	Menampilkan halaman dari daftar Request Order
8	Tombol Tambah Request Order	Tombol yang digunakan untuk menambah Request Order baru dengan menampilkan form Request Order

16. Halaman Detail Request Order

Halaman Detail Request Order adalah halaman yang berisi detail dari bahan baku dan bisa diubah sesuai keinginan. Halaman Detail Request Order akan dijelaskan pada gambar 5.22 dan penjelasan detail akan dijelaskan pada table 5.16.

Gambar 5.22 Tampilan Antarmuka Halaman Detail Request Order

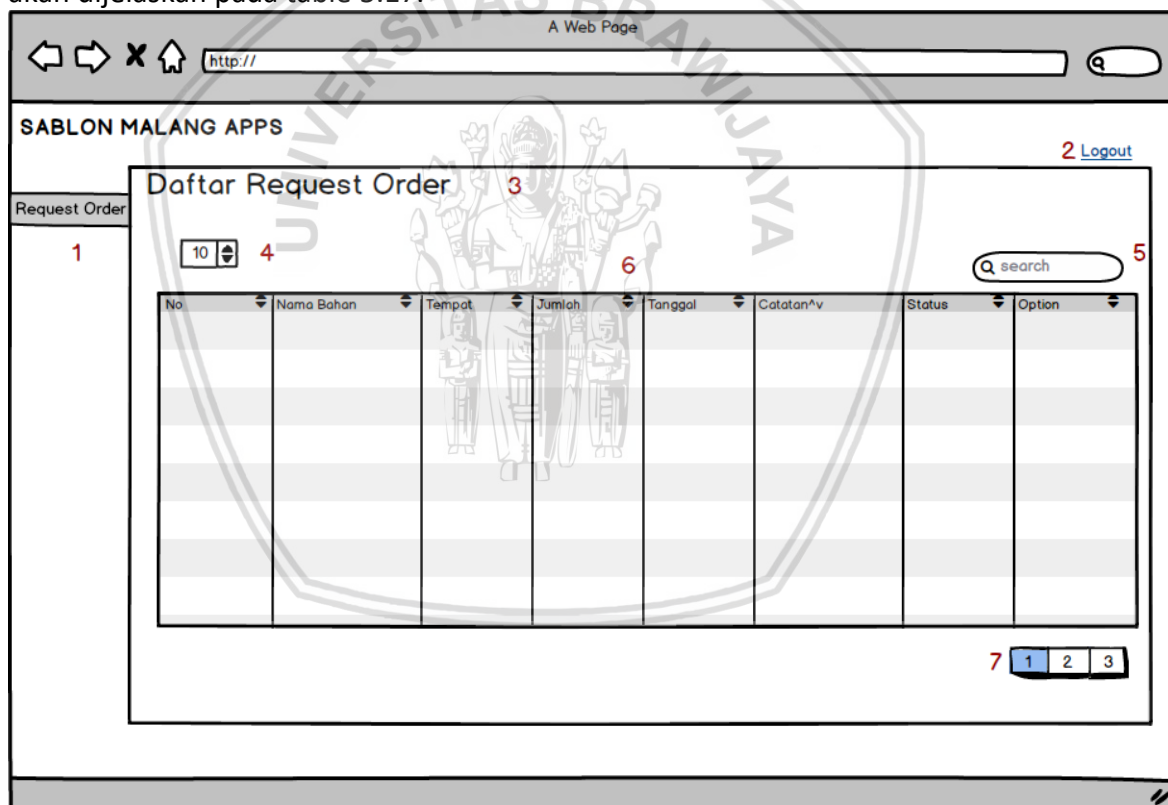
Table 5.16 Penjelasan Antarmuka Halaman Detail Request Order

No	Nama Object	Keterangan
----	-------------	------------

1	Navbar	Menampilkan pilihan menu Daftar Request Order
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Detail Request Order
4	Form detail Request Order	Menampilkan form detail Request Order yang dapat diubah
5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form detail Request Order yang sudah dirubah

17. Halaman Validasi Request Order

Halaman Validasi Request Order adalah halaman yang berisi daftar Request Order yang sudah terdaftar baik itu konveksi maupun sablon sesuai hak akses. Halaman Validasi Request Order akan dijelaskan pada gambar 5.23 dan penjelasan detail akan dijelaskan pada table 5.17.



Gambar 5.23 Tampilan Antarmuka Halaman Validasi Request Order

Table 5.17 Penjelasan Antarmuka Halaman Validasi Request Order

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Request Order
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem

3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Request Order
4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar bahan baku yang tersedia
6	Tabel Daftar Request Order	Menampilkan semua Daftar Request Order dan berisi opsi untuk menyetujui dan menolak
7	Button Angka	Menampilkan halaman dari daftar Request Order

18. Halaman Tambah Jadwal Induk Produksi

Halaman Tambah Jadwal Induk Produksi adalah halaman yang form untuk menambahkan Jadwal Induk Produksi setiap periodenya. Halaman Tambah Jadwal Induk Produksi akan dijelaskan pada gambar 5.24 dan penjelasan detail akan dijelaskan pada table 5.18.

Gambar 5.24 Tampilan Antarmuka Halaman Tambah Jadwal Induk Produksi

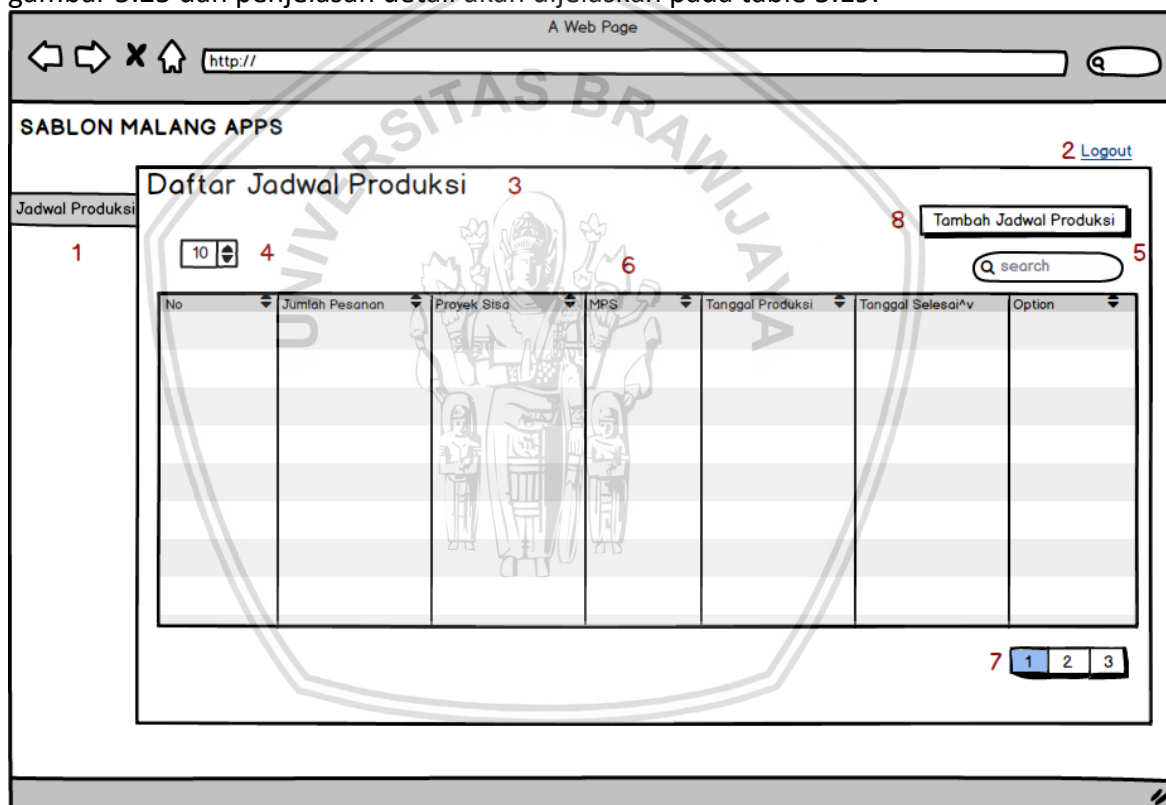
Table 5.18 Penjelasan Antarmuka Halaman Tambah Jadwal Induk Produksi

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Jadwal Produksi

2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Tambah Jadwal Induk Produksi
4	Form Jadwal Induk Produksi	Menampilkan form Jadwal Induk Produksi
5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form Jadwal Induk Produksi yang sudah dimasukkan

19. Halaman Daftar Jadwal Induk Produksi

Halaman Daftar Jadwal Induk Produksi adalah halaman yang berisi daftar rencana produksi setiap periode. Halaman Daftar Jadwal Induk Produksi akan dijelaskan pada gambar 5.25 dan penjelasan detail akan dijelaskan pada table 5.19.



Gambar 5.25 Tampilan Antarmuka Halaman Daftar Jadwal Induk Produksi

Table 5.19 Penjelasan Antarmuka Halaman Daftar Jadwal Induk Produksi

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Jadwal Produksi
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Jadwal Induk Produksi

4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar Jadwal Induk Produksi yang tersedia
6	Tabel Daftar Pesanan	Menampilkan semua Daftar Jadwal Induk Produksi yang telah diolah
7	Button Angka	Menampilkan halaman dari daftar Jadwal Induk Produksi
8	Tombol Tambah Jadwal Produksi	Tombol yang berfungsi untuk menambah Jadwal Produksi baru dengan menampilkan form jadwal produksi

20. Halaman Detail Jadwal Induk Produksi

Halaman Detail Jadwal Induk Produksi adalah halaman yang form untuk mengubah Jadwal Induk Produksi yang sudah berada di database dengan yang baru. Halaman Detail Jadwal Induk Produksi akan dijelaskan pada gambar 5.26 dan penjelasan detail akan dijelaskan pada table 5.20.

Gambar 5.26 Tampilan Antarmuka Halaman Detail Jadwal Induk Produksi

Table 5.20 Penjelasan Antarmuka Halaman Detail Jadwal Induk Produksi

No	Nama Object	Keterangan
----	-------------	------------

1	Navbar	Menampilkan pilihan menu Ubah Jadwal Produksi
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Ubah Jadwal Induk Produksi
4	Form Jadwal Induk Produksi	Menampilkan form Jadwal Induk Produksi
5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form Jadwal Induk Produksi dengan data yang baru

21. Halaman Tambah Pembelian

Halaman Tambah Pembelian adalah halaman yang berisi form untuk menambahkan Pembelian baru. Halaman Tambah Request Order akan dijelaskan pada gambar 5.27 dan penjelasan detail akan dijelaskan pada table 5.21.

The screenshot shows a web browser window with the address bar displaying 'http://'. The page title is 'SABLON MALANG APPS'. The main content area is titled 'Tambah Pembelian Bahan Baku'. The form includes a date picker for 'Tanggal', a text area for 'Catatan', a dropdown menu for 'Nama Bahan', and a text input for 'Jumlah (KG)'. At the bottom, there are two buttons: 'Tambah Bahan' and 'Submit'. The page is numbered 1, 2, 3, 4, 5, and 6, corresponding to the table above.

Gambar 5.27 Tampilan Antarmuka Halaman Tambah Pembelian

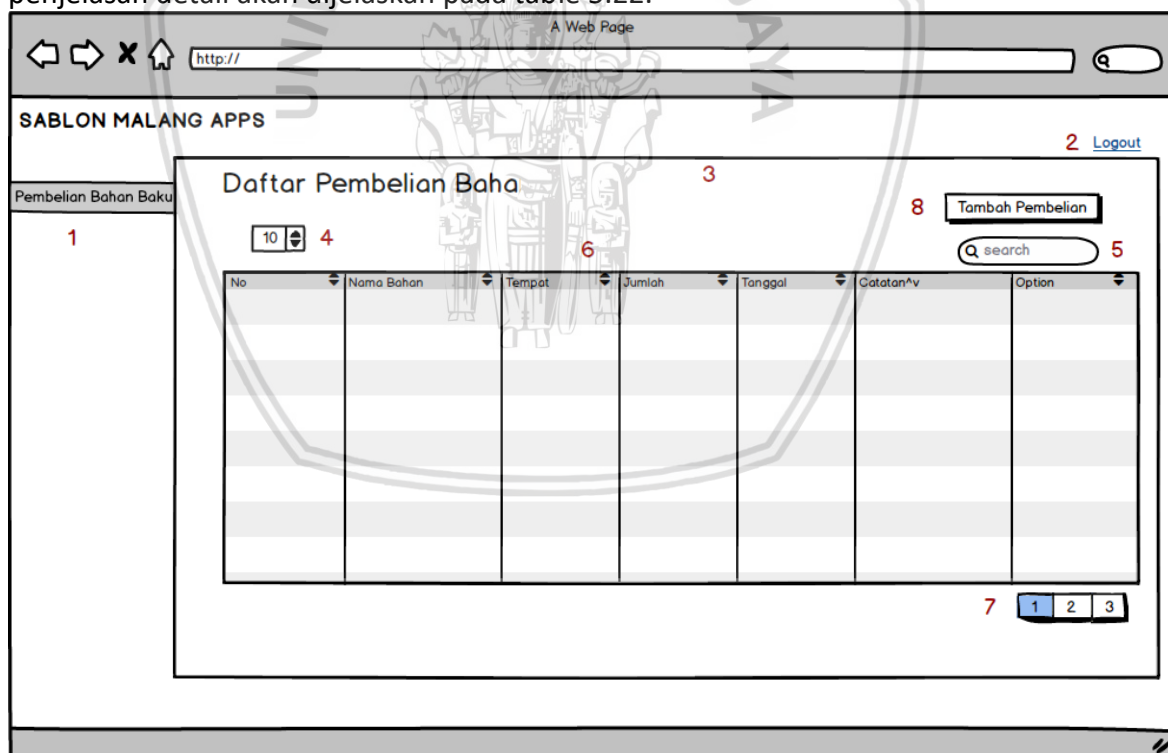
Table 5.21 Penjelasan Antarmuka Halaman Tambah Pembelian

No	Nama Object	Keterangan
----	-------------	------------

1	Navbar	Menampilkan pilihan menu Tambah Tambah Pembelian
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Tambah Pembelian
4	Form Tambah Pembelian	Menampilkan form Tambah Pembelian yang akan diisi secara detail
5	Tombol Tambah	Tombol yang digunakan untuk menambah form Tambah Pembelian
6	Tombol Submit	Tombol yang berfungsi untuk menyimpan form Tambah Pembelian yang nanti akan ditambahkan kedalam

22. Halaman Daftar Pembelian

Halaman Daftar Pembelian adalah halaman yang berisi daftar Pembelian yang sudah tersimpan. Halaman Daftar Pembelian akan dijelaskan pada gambar 5.28 dan penjelasan detail akan dijelaskan pada table 5.22.



Gambar 5.28 Tampilan Antarmuka Halaman Daftar Pembelian

Table 5.22 Penjelasan Antarmuka Halaman Daftar Pembelian

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Pembelian

2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar Pembelian
4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari daftar Pembelian yang tersedia
6	Tabel Daftar Pembelian	Menampilkan semua Daftar Pembelian sesuai dengan namaUser dan berisi opsi untuk mengelola Pembelian tersebut
7	Button Angka	Menampilkan halaman dari daftar Pembelian
8	Tombol Tambah Pembelian	Tombol yang digunakan untuk menambah Pembelian baru dengan menampilkan form Pembelian

23. Halaman Detail Pembelian

Halaman Detail Pembelian adalah halaman yang berisi detail dari Pembelian dan bisa diubah sesuai keinginan. Halaman Detail Pembelian akan dijelaskan pada gambar 5.29 dan penjelasan detail akan dijelaskan pada table 5.23.

Gambar 5.29 Tampilan Antarmuka Halaman Detail Pembelian

Table 5.23 Penjelasan Antarmuka Halaman Detail Pembelian

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar Pembelian
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Detail Pembelian
4	Form detail Pembelian	Menampilkan form detail Pembelian yang dapat diubah
5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form detail Pembelian yang sudah dirubah

24. Halaman Tambah User

Halaman Tambah User adalah halaman yang form untuk menambahkan user. Halaman Tambah User akan dijelaskan pada gambar 5.30 dan penjelasan detail akan dijelaskan pada table 5.24.

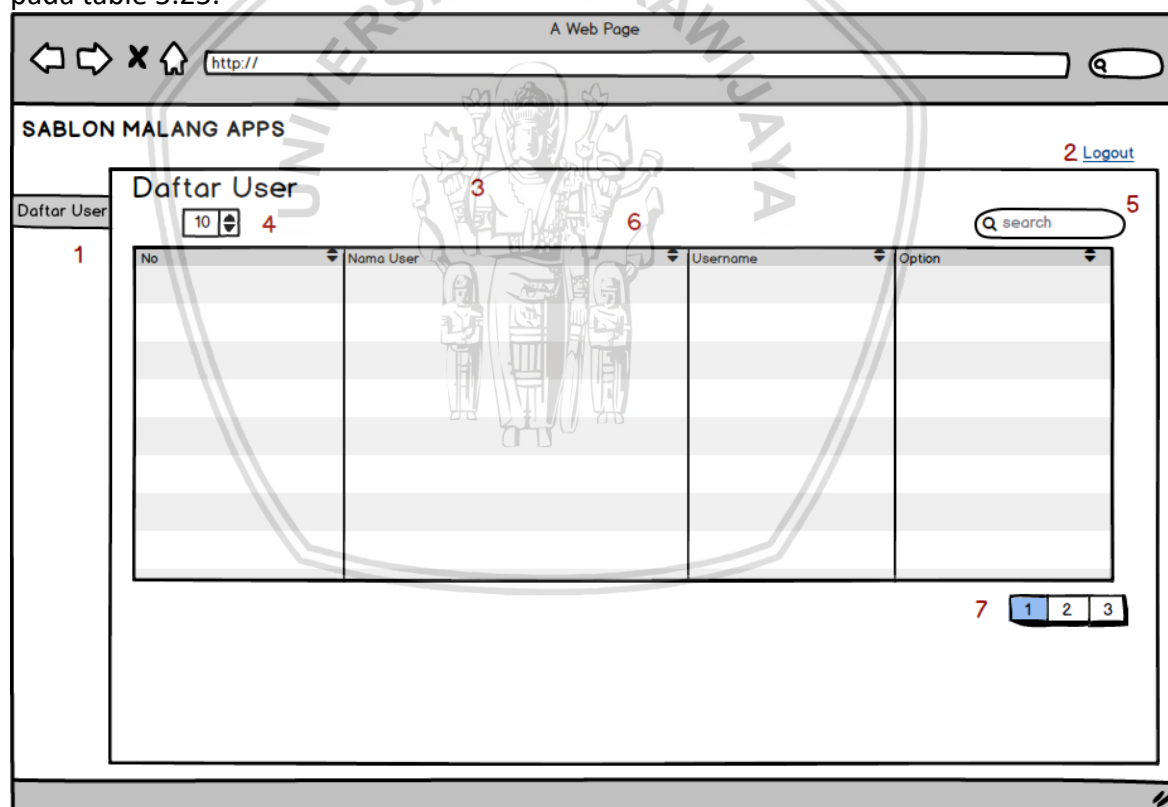
Gambar 5.30 Tampilan Antarmuka Halaman Tambah User

Table 5.24 Penjelasan Antarmuka Halaman Tambah User

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Tambah User
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu User baru
4	Form User	Menampilkan form User
5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form User yang sudah dimasukkan

25. Halaman Daftar User

Halaman Daftar User adalah halaman yang berisi daftar data semua user. Halaman Daftar User akan dijelaskan pada gambar 5.31 dan penjelasan detail akan dijelaskan pada table 5.25.

**Gambar 5.31 Tampilan Antarmuka Halaman Daftar User****Table 5.25 Penjelasan Antarmuka Halaman Daftar User**

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar User

2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Daftar User
4	Nomor baris	Jumlah yang nanti akan ditampilkan dalam baris tabel
5	Menu Search	Menu yang berfungsi untuk mencari data User yang tersedia
6	Tabel Daftar Pesanan	Menampilkan semua Daftar User
7	Button Angka	Menampilkan halaman dari daftar User

26. Halaman Detail User

Halaman Detail User adalah halaman untuk Mengubah data user. Halaman Detail User akan dijelaskan pada gambar 5.32 dan penjelasan detail akan dijelaskan pada table 5.26.

The screenshot shows a web browser window with the address bar displaying 'http://'. The page title is 'SABLON MALANG APPS'. The main content area is titled 'Detail User'. On the left, there is a sidebar with a link 'Daftar User' labeled '1'. The main form area contains the following elements: a 'Logout' link labeled '2' in the top right; the form title 'Detail User' labeled '3'; a 'Nama User' dropdown menu; text input fields for 'username', 'password', and 'password baru'; and a 'Submit' button labeled '5'. A large watermark of the Universitas Brawijaya logo is visible in the background.

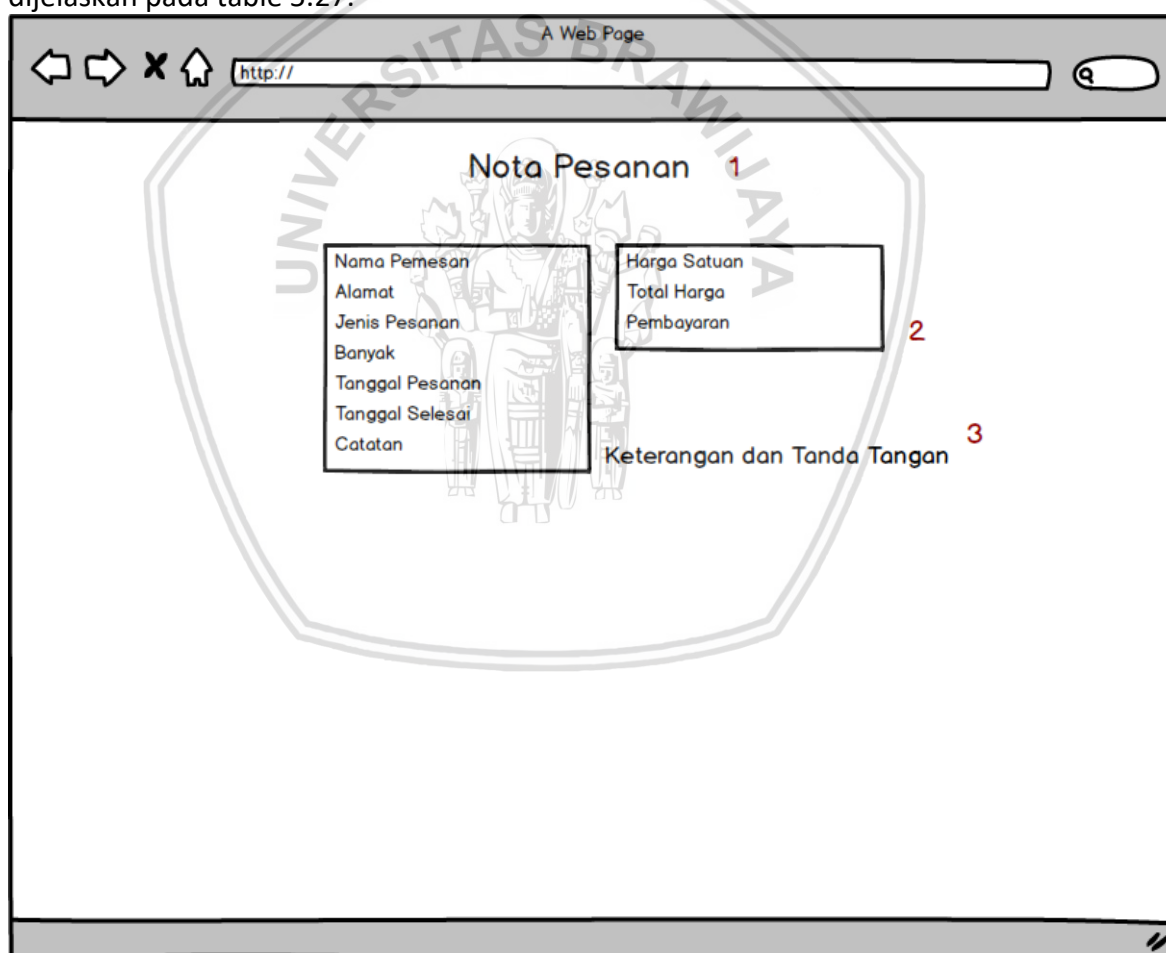
Gambar 5.32 Tampilan Antarmuka Halaman Detail User

Table 5.26 Penjelasan Antarmuka Halaman Detail User

No	Nama Object	Keterangan
1	Navbar	Menampilkan pilihan menu Daftar User
2	Tombol Logout	Tombol yang bertujuan untuk keluar dari sistem
3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Detail User
4	Form User	Menampilkan form User
5	Tombol Submit	Tombol yang berfungsi untuk menyimpan form User baru yang sudah dimasukkan.

27. Halaman Mencetak Nota

Halaman Mencetak Nota adalah halaman untuk mencetak data pesanan. Halaman Mencetak Nota akan dijelaskan pada gambar 5.33 dan penjelasan detail akan dijelaskan pada table 5.27.



Gambar 5.33 Tampilan Antarmuka Halaman Mencetak Nota

Table 5.27 Penjelasan Antarmuka Halaman Mencetak Nota

No	Nama Object	Keterangan
----	-------------	------------

3	Judul Halaman	Menampilkan nama dari halaman tersebut yaitu Nota Pesanan
4	Form Nota	Menampilkan form nota berupa data pesanan
5	Tulisan Keterangan	Menampilkan keterangan dari form nota

5.2 Implementasi

Setelah proses perancangan selesai dilakukan tahapan selanjutnya adalah melakukan proses implementasi. Implementasi dilakukan berdasarkan hasil yang didapat dari rekayasa kebutuhan dan perancangan. Setiap kebutuhan yang sudah didefinisikan akan diimplementasikan pada sistem. Tahap implementasi akan menjelaskan mengenai spesifikasi sistem, batasan implementasi, implementasi *class*, implementasi kode program, dan implementasi antarmuka.

5.2.1 Spesifikasi Sistem

Proses implementasi pada sistem manajemen bahan baku dan pemesanan pada sablon malang dibantu oleh perangkat keras dan perangkat lunak sehingga dapat mewujudkan sistem ini. Pada tahap spesifikasi sistem akan menjelaskan tentang spesifikasi dari perangkat keras dan perangkat lunak yang digunakan.

5.2.1.1 Spesifikasi Perangkat Keras

Pembuatan sistem manajemen bahan baku dan pemesanan pada sablon malang menggunakan perangkat keras dengan spesifikasi yang dijelaskan pada tabel 5.28.

Tabel 5.28 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
<i>System Model</i>	Toshiba Satellite L745
<i>Processor</i>	Intel® Core™ i3-2350M
<i>Memory</i>	4096MB
<i>Display</i>	Intel® HD Graphics
<i>Harddisk</i>	500 GB

5.2.1.2 Spesifikasi Perangkat Lunak

Pembuatan sistem manajemen bahan baku dan pemesanan pada sablon malang menggunakan perangkat lunak dengan spesifikasi yang akan dijelaskan pada tabel 5.29.

Tabel 5.29 Spesifikasi Perangkat Lunak

Nama Komponen	Spesifikasi
<i>Operating System</i>	Windows 10 Profesional 64-bit

Programming Language	PHP, JavaScript
Programming Environment	Sublime 3
Datavase Manageny System	MySQL 5.6.3

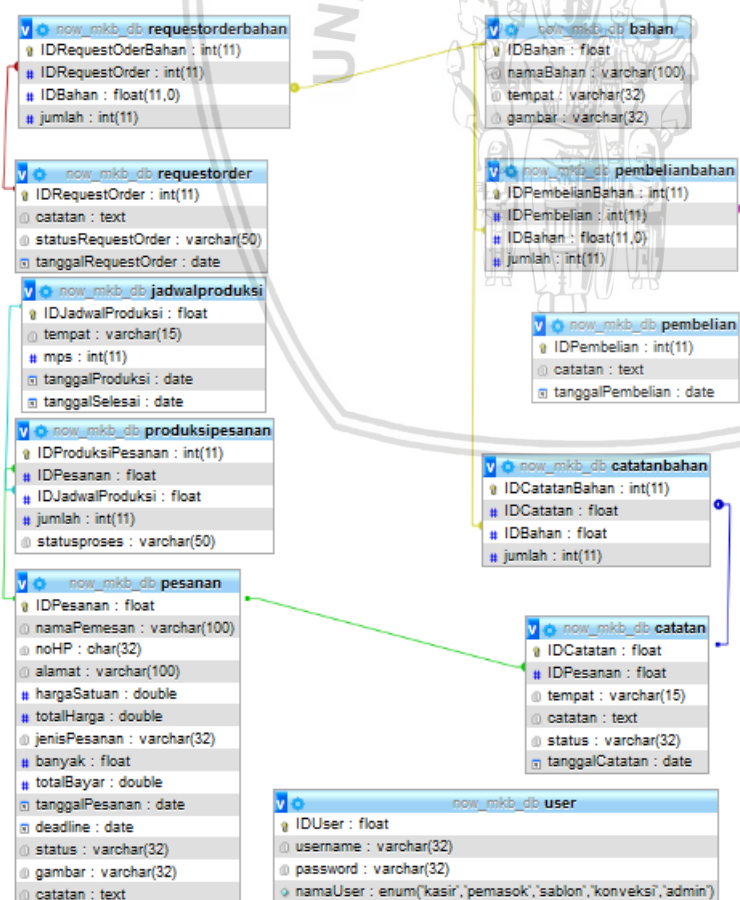
5.2.2 Batasan Implementasi

Terdapat batasan-batasan pada proses implementasi sistem manajemen bahan baku dan pemesana pada sablon malang yaitu sebagai berikut :

1. Aplikasi dibangun berbasis *web* menggunakan bahasa pemrograman PHP dan JavaScript
2. *Framework* yang digunakan adalah *CodeIgniter*
3. *Database* dibangun menggunakan *MySQL*
4. Pembuatan layout pada antarmuka aplikasi menggunakan HTML, CSS, dan JavaScript

5.2.3 Implementasi Basis Data

Implementasi basis data dibuat berdasarkan perancangan database secara fisik yaitu *physical data model* yang sudah dibuat pada bab perancangan. Implementasi basis data digambarkan pada gambar 5.34.



Gambar 5.34 Implementasi Database

5.2.4 Implementasi Class

Proses implementasi *class* dilakukan berdasarkan perancangan *class diagram*. Klas-klas yang ada pada *class diagram* akan direalisasikan ke dalam bentuk *file* dengan format (.php). Hasil implementasi akan ditunjukkan pada tabel 5.30

Tabel 5.30 Implementasi Class Diagram

No	Package	Nama Class	Nama File
1	Controller	BahanBakuBaru	BahanBakuBaru.php
2	Controller	CatatanBaru	CatatanBaru.php
3	Controller	CatatanPesanan	CatatanPesanan.php
4	Controller	DaftarBahan	DaftarBahan.php
5	Controller	DaftarPembelian	DaftarPembelian.php
6	Controller	DaftarPersediaan	DaftarPersediaan.php
7	Controller	DaftarPesanan	DaftarPesanan.php
8	Controller	Kasir	Kasir.php
9	Controller	Konveksi	Konveksi.php
10	Controller	Login	Login.php
11	Controller	Gudang	Gudang.php
12	Controller	PesananBaru	PesananBaru.php
13	Controller	Sablon	Sablon.php
14	Controller	JadwalProduksiBaru	JadwalProduksiBaru.php
15	Controller	JadwalProduksi	LaporanJadwalProduksi.php
16	Controller	RequestOrder	RequestOrder.php
17	Controller	Admin	Admin.php
18	View	daftarBahanBaku	daftarBahanBaku.php
19	View	daftarCatatan	daftarCatatan.php
20	View	daftarJadwalProduksi	daftarJadwalProduksi.php
21	View	daftarPembelian	daftarPembelian.php
22	View	daftarPersediaan	daftarPersediaan.php
23	View	detailPesanan	detailPesanan.php
24	View	daftarRequestOrder	daftarRequestOrder.php
25	View	daftarUser	daftarUser.php
26	View	detailBahan	detailBahan.php

27	View	detailJadwalProduksi	detailJadwalProduksi.php
28	View	detailPesanan	detailPesanan.php
29	View	footer	footer.php
30	View	formBahan	formBahan.php
31	View	formCatatan	formCatatan.php
32	View	formEditCatatan	formEditCatatan.php
33	View	formEditPembelian	formEditPembelian.php
34	View	formEditPesanan	formEditPesanan.php
35	View	formEditRequestOrder	formEditRequesstOrder.php
36	View	formEditUser	formEditUser.php
36	View	formJadwalProduksi	formJadwalProduksi.php
37	View	formLogin	formLogin.php
38	View	formPembelian	formPembelian.php
39	View	formPesanan	formPesanan.php
40	View	formRequestOrder	formRequestOrder.php
41	View	formUser	formUser.php
42	View	Header	Header.php
43	View	jadwalProduksiPesanan	jadwalProduksiPesanan.php
44	View	notaPesanan	notaPesanan.php
45	Model	Bahan	Bahan.php
46	Model	Catatan	Catatan.php
47	Model	M_login	M_login.php
48	Model	Pesanan	Pesanan.php
49	Model	M_RequestOrder	M_RequestOrder.php
50	Model	M_JadwalProduksi	M_JadwalProduksi.php
51	Model	Pembelian	Pembelian.php

5.2.5 Implementasi Kode Program

Proses implementasi kode program dilakukan berdasarkan perancangan algoritma. Tahap implementasi ini akan menunjukkan perubahan algoritma menjadi kode program atau *source code* yang dapat dimengerti oleh komputer. Algoritma yang telah dibuat akan diubah menjadi kode program menggunakan Bahasa pemrograman PHP. Implementasi kode program akan ditunjukkan pada beberapa sampel operasi yaitu prosesTambah() pada

klas PesananBaru, operasi tampilkanDaftarBahan() pada klas DaftarBahan, dan operasi prosesTambah() pada CatatanBaru.

Nama Klas : CatatanVaru

Nama Operasi : prosesCek

Kode Program :

```

public function prosesCek($tempat, $id){
    if ($post = $this->input->post()) {
        $message = false;
        $temp_data_bahan = $this->pembelian->getPersediaan()-
>result();

        $data_bahan = array();
        foreach ($temp_data_bahan as $row) {
            $data_bahan[$row->IDBahan] = $row;
        }
        $data_bahan_cek = array();
        foreach ($post['bahan'] as $key => $value) {
            if (isset($data_bahan_cek[$value])) {
                $data_bahan_cek[$value]['jumlah'] +=
$post['jumlah'][$key];
            }else{
                $data_bahan_cek[$value] = array(
                    'IDBahan' => $value,
                    'jumlah' =>
$post['jumlah'][$key],
                );
            }
        }
        foreach ($data_bahan_cek as $key => $value) {
            if (isset($data_bahan[$key])) {
                if ($data_bahan[$key]->jumlah_persediaan <
$value['jumlah']) {
                    $message = "Jumlah bahan melebihi
persediaan.";
                    break;
                }
            }
        }
    }
}

```

```

        }else{
            $message = "Data bahan tidak ditemukan.";
            break;
        }
    }

    if (!$message) {
        $post['IDPesanan'] = $id;
        $post['status'] = 'temp';
        $post['tempat'] = $tempat;
        $id_insert = $this->catatan->tambah($post);
        $query = $this->pesanan->getDetail($id);
        $new_temp = array();
        foreach ($query->result() as $key => $value) {
            if ($value->tempat == $tempat) {
                $new_temp[] = $value->IDProduksiPesanan;
            }
        }
        $this->db->where('IDPesanan', $id);
        $this->db->where_in('IDProduksiPesanan', $new_temp);
        $this->db->update('produksipesanan', array('statusproses'
=> 'Proses'));

        $new_data = array();
        foreach ($post['bahan'] as $key => $value) {
            if (isset($new_data[$value])) {
                $new_data[$value]['jumlah'] += $post['jumlah'][$key];
            } else {
                $new_data[$value] = array(
                    'IDCatatan' => $id_insert,

```



```

                                'IDBahan'                =>
$value,

                                'jumlah'                =>
$post['jumlah'][$key],

                                );

                                }

                                }
                                $this->db->insert_batch('catatanbahan', $new_data);
                                redirect('daftarPesanan/detail/'.$id);
                            }else{
                                $this->cek($tempat, $id, $message);
                            }
                        }
                    }
                }
            }
        }
    }
}

```

Nama Klas : RequestOrder
 Nama Operasi : prosesTambah
 Kode Program :

```

public function prosesTambah(){
    if ($post = $this->input->post()) {
        $id = $this->m_requestOrder->tambah($post);
        $new_data = array();
        for ($i=0; $i < count($post['IDBahan']); $i++) {
            $new_data[] = array(
                'IDRequestOrder' => $id,
                'IDBahan' => $post['IDBahan'][$i],
                'jumlah' => $post['jumlah'][$i],
            );
        }
        $this->db->insert_batch('requestorderbahan', $new_data);
        redirect('requestOrder');
    }
    $this->tambah();
}

```

```
}

```

Nama Klas : JadwalProduksiBaru

Nama Operasi : prosesTambah

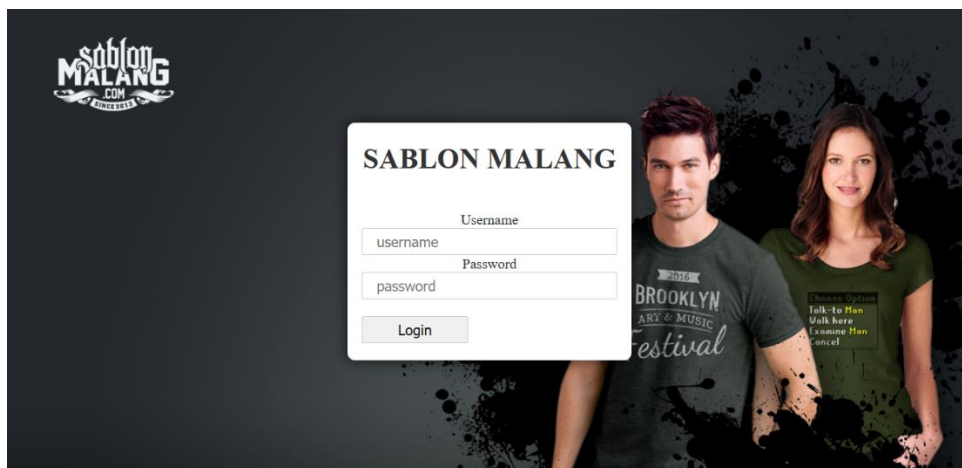
Kode Program :

```
public function prosesTambah(){
    if ($post = $this->input->post()) {
        if (protected_item('konveksi')) {
            $post['tempat'] = 'konveksi';
        }
        else{
            $post['tempat'] = 'sablon';
        }
        $this->m_jadwalproduksi->tambah($post);
        redirect('jadwalproduksi');
    }
    redirect('jadwalproduksi');
}
```

5.2.6 Implementasi Antarmuka

Implementasi antarmuka dilakukan berdasarkan perancangan antarmuka. Berikut adalah halaman antarmuka yang terdapat pada system manajemen bahan baku dan pemesanan sablon malang.

5.2.6.1 Implementasi Halaman Login



Gambar 5.35 Implementasi Halaman Login

Gambar 5.35 merupakan tampilan pertama kali diakses oleh user yaitu tampilan login yang berfungsi untuk melanjutkan kedalam system berdasarkan username. Tampilan antarmuka halaman login dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.2 Implementasi Halaman Tambah Pesanan

Pesanan Baru

☒ Forms

Nama Pemesan
Fakhruzzahid

Nomor HP
085141270696

Alamat
Malang

Jenis Pesanan
Konveksi dan Sablon

Banyak
12

Harga Satuan
100000

Total Harga
1200000

Pembayaran
100000

Tanggal Pesanan
07/01/2018

Tanggal Selesai
07/24/2018

Catatan
Kain Hitam dengan Benang Putih

Gambar
Browse... IMG_6348c.jpg

Jadwal Produksi Konveksi
Tambah Jadwal

Jadwal Produksi Sablon
Tambah Jadwal

Jadwal Produksi
09 Jul 2018 - 13 Jul 2018 (Sisa Produksi: 50)

Banyak
12

Hapus Jadwal

Jadwal Produksi
16 Jul 2018 - 20 Jul 2018 (Sisa Produksi: 40)

Banyak
12

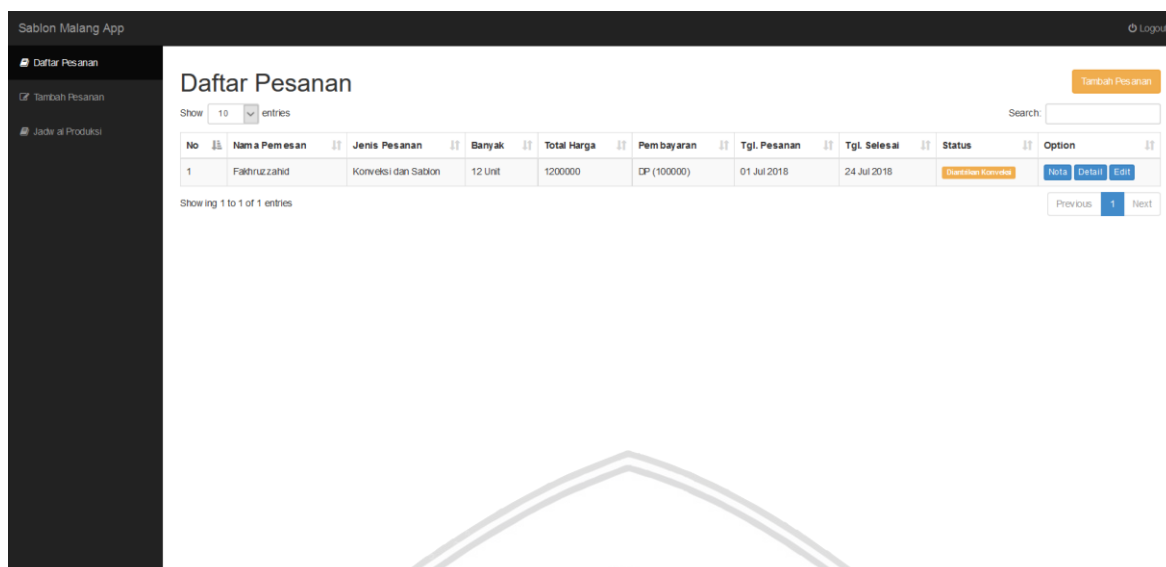
Hapus Jadwal

Submit

Gambar 5.36 Implementasi Halaman Tambah Pesanan

Gambar 5.36 merupakan tampilan form pesanan yang hanya bisa diakses oleh kasir. Tampilan antarmuka halaman Tambah Pesanan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

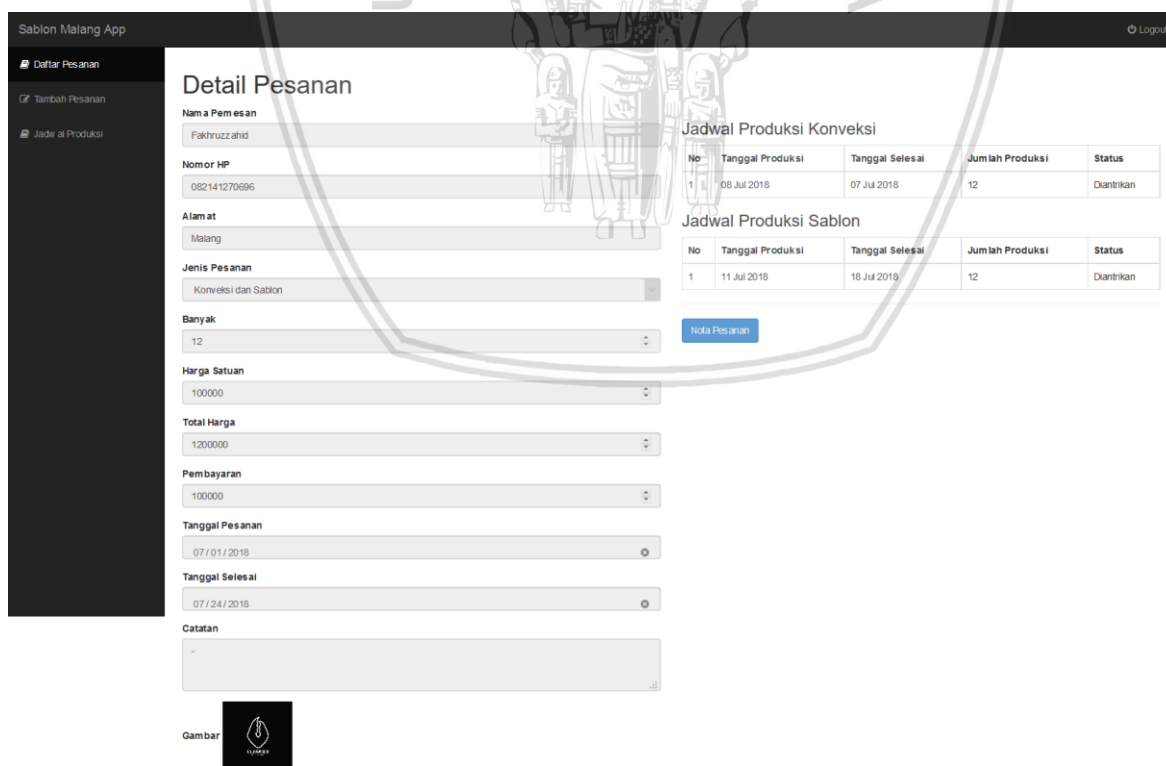
5.2.6.3 Implementasi Halaman Daftar Pesanan



Gambar 5.37 Implementasi Halaman Daftar Pesanan

Gambar 5.37 merupakan tampilan tabel dari semua pesanan yang sudah berada didalam database. Tampilan antarmuka halaman Daftar Pesanan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.4 Implementasi Halaman Detail Pesanan



Gambar 5.38 Implementasi Halaman Detail Pesanan

Gambar 5.38 merupakan tampilan detail pesanan berdasarkan pesanan yang sudah dipilih dan tidak bias diubah. Halaman ini bisa diakses ketika memilih lihat detail yang berada pada daftar pesanan. Tampilan antarmuka halaman Detail Pesanan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.5 Implementasi Halaman Ubah Pesanan

Sablon Malang App Logout

Detail Pesanan

Nama Pemesan
Fakhruzzahid

Nomor HP
082141270696

Alamat
Malang

Jenis Pesanan
Konveksi dan Sablon

Banyak
12

Harga Satuan
100000

Total Harga
1200000

Pembayaran
100000

Tanggal Pesanan
07/01/2018

Tanggal Selesai
07/24/2018

Catatan
-

Gambar

Submit

Jadwal Produksi Konveksi

No	Tanggal Produksi	Tanggal Selesai	Jumlah Produksi	Status
1	08 Jul 2018	07 Jul 2018	12	Diantrikan

Jadwal Produksi Sablon

No	Tanggal Produksi	Tanggal Selesai	Jumlah Produksi	Status
1	11 Jul 2018	18 Jul 2018	12	Diantrikan

Gambar 5.39 Implementasi Halaman Ubah Pesanan

Gambar 5.39 merupakan tampilan detail pesanan berdasarkan pesanan yang sudah dipilih dan bisa diubah. Halaman ini bisa diakses ketika memilih tombol edit yang berada pada daftar pesanan. Tampilan antarmuka halaman Ubah Pesanan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.6 Implementasi Halaman Jadwal Produksi Pesanan

Gambar 5.40 Implementasi Halaman Jadwal Produksi Pesanan

Gambar 5.40 merupakan tampilan Jadwal Produksi Pesanan. Halaman ini bisa diakses ketika memilih tombol pesanan yang berada pada jadwal produksi. Tampilan antarmuka halaman jadwal produksi pesanan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.7 Implementasi Halaman Check Persediaan

Gambar 5.41 Implementasi Halaman Check Persediaan

Gambar 5.41 merupakan tampilan form dari cek persediaan. Halaman ini bisa diakses ketika memilih tombol detail pada pesanan di jadwal produksi pesanan kemudian memilih tombol check persediaan. Tampilan antarmuka halaman Check Persediaan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.8 Implementasi Halaman Tambah Catatan

Catatan Pesanan

Form

Jika bahan baku tidak mencukupi silahkan request order.

Tanggal Catatan: 07/08/2018

Catatan: -

Nama Bahan: Kain Hitam (Persediaan: 12)

Jumlah (KG): 4

Submit

Detail Pesanan

Nama Pemesan: Fakhruzzahid

Banyak: 12

Catatan: -

Gambar:

Gambar 5.42 Implementasi Halaman Tambah Catatan

Gambar 5.42 merupakan tampilan form dari tambah Catatan. Halaman ini bisa diakses ketika memilih tombol detail pada pesanan di jadwal produksi pesanan kemudian memilih tombol tambah catatan. Tampilan antarmuka halaman Tambah Catatan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.9 Implementasi Halaman Daftar Catatan

Daftar Catatan

Show 10 entries

No	Nama Pemesan	Nama Bahan	Jumlah (KG)	Tanggal Catatan	Catatan
1	Fakhruzzahid	Kain Hitam	4	08 Jul 2018	-

Showing 1 to 1 of 1 entries

Previous 1 Next

Gambar 5.43 Implementasi Halaman Daftar Catatan

Gambar 5.43 merupakan tampilan dari daftar catatan yang sudah selesai disimpan. Tampilan antarmuka halaman Daftar Catatan dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.10 Implementasi Halaman Tambah Bahan Baku

Gambar 5.44 Implementasi Halaman Tambah Bahan baku

Gambar 5.44 merupakan tampilan form Tambah Bahan Baku. Tampilan antarmuka halaman Tambah Bahan Baku dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.11 Implementasi Halaman Daftar Bahan Baku

No	Nama Bahan	Tempat	Gambar	Option
1	Kain Hitam	Konveksi		Edit
2	Tinta Hijau	sablon		Edit
3	Tinta Kuning	sablon		Edit
4	Kain Bamboo Putih	Konveksi		Edit

Gambar 5.46 Implementasi Halaman Daftar Bahan Baku

Gambar 5.45 merupakan tampilan tabel dari semua bahan baku. Tampilan antarmuka halaman Daftar Bahan Baku dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.12 Implementasi Halaman Detail Bahan Baku

Gambar 5.46 Implementasi Halaman Detail Bahan Baku

Gambar 5.46 merupakan tampilan detail Bahan Baku berdasarkan bahan baku yang sudah dipilih dan bisa diubah. Halaman ini bisa diakses ketika memilih tombol Edit yang berada pada daftar pesanan. Tampilan antarmuka halaman Detail Bahan Baku dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.13 Implementasi Halaman Daftar Persediaan Bahan baku

No	Nama Bahan	Gambar	Jumlah
1	Kain Hitam		8

Gambar 5.47 Implementasi Halaman Daftar Persediaan Bahan baku

Gambar 5.47 merupakan tampilan tabel dari semua persediaan bahan baku. Tampilan antarmuka halaman Daftar Persediaan Bahan Baku dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.14 Implementasi Halaman Tambah Request Order

Sablon Malang App

Logout

Jadwal Produksi

Tambah Jadwal Produksi

Catatan Pesanan

Persediaan Bahan Baku

Request Order

Tambah Request Order

Forms

Tanggal
mm/dd/yyyy

Catatan
Catatan

Nama Bahan
Kain Hitam

Jumlah
Jumlah Bahan Baku

Hapus Bahan

Tambah Bahan Submit

Gambar 5.48 Implementasi Halaman Request Order

Gambar 5.48 merupakan tampilan form Request Order. Tampilan antarmuka halaman Tambah Request Order dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.15 Implementasi Halaman Daftar Request Order

Sablon Malang App

Logout

Jadwal Produksi

Tambah Jadwal Produksi

Catatan Pesanan

Persediaan Bahan Baku

Request Order

Daftar Request Order

Tambah Request Order

No	Nama Bahan	Tempat	Jumlah	Tanggal	Catatan	Status	Option
1	Tinta Kuning	sablon	13	03 Juli 2018	-	Diajukan	Edit
No	Nama Bahan	Tempat	Jumlah	Tanggal	Catatan	Status	Option

Gambar 5.49 Implementasi Halaman Daftar Request Order

Gambar 5.49 merupakan tampilan tabel dari semua Request Order. Tampilan antarmuka halaman Daftar Request Order dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.16 Implementasi Halaman Detail Request Order

Gambar 5.50 Implementasi Halaman Detail Request Order

Gambar 5.50 merupakan tampilan detail Request Order berdasarkan Request Order yang sudah dipilih dan bisa diubah. Halaman ini bisa diakses ketika memilih tombol Edit yang berada pada daftar Request Order. Tampilan antarmuka halaman Detail Request Order dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.17 Implementasi Halaman Validasi Request Order

No	Nama Bahan	Tempat	Jumlah	Tanggal	Catatan	Status	Option
1	Tinta Kuning	sablon	13	03 Juli 2018	-	Dijukan	Setujui Tolak

Gambar 5.51 Implementasi Halaman Validasi Request Order

Gambar 5.51 merupakan tampilan Halaman Validasi Request Order yang hanya bisa diakses oleh Gudang. Tampilan antarmuka halaman Detail Request Order dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.18 Implementasi Halaman Tambah Jadwal Induk Produksi

Sablon Malang App

Jadwal Produksi

Tambah Jadwal Produksi

Catatan Pesanan

Persediaan Bahan Baku

Request Order

Tambah Jadwal Produksi

Tanggal Produksi

07/08/2018

Tanggal Selesai

07/07/2018

MPS

Jumlah MPS

Submit

Gambar 5.52 Implementasi Halaman Tambah Jadwal Produksi

Gambar 5.52 merupakan tampilan form Tambah Jadwal Produksi. Tampilan antarmuka halaman Tambah Jadwal Produksi dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.19 Implementasi Halaman Daftar Jadwal Produksi

Sablon Malang App

Jadwal Produksi

Tambah Jadwal Produksi

Catatan Pesanan

Persediaan Bahan Baku

Request Order

Tambah Jadwal

Daftar Jadwal Induk Produksi

Show 10 entries

No	Jumlah Pesanan	Proyek Sisa	MPS	Tanggal Produksi	Tanggal Selesai	Option
1	0	30	30	01 Jul 2018	07 Jul 2018	Edit Pesanan
2	0	40	40	08 Jul 2018	07 Jul 2018	Edit Pesanan
No	Jumlah Pesanan	Proyek Sisa	MPS	Tanggal Produksi	Tanggal Selesai	Option

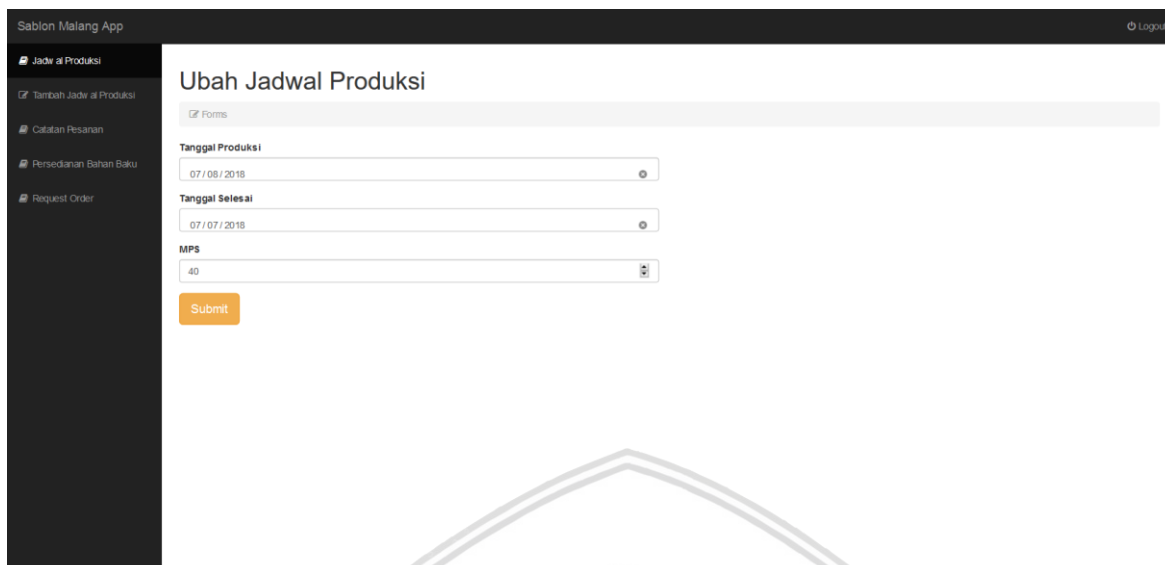
Showing 1 to 2 of 2 entries

Previous 1 Next

Gambar 5.53 Implementasi Halaman Daftar Jadwal Produksi

Gambar 5.53 merupakan tampilan tabel dari semua jadwal produksi. Tampilan antarmuka halaman Daftar Jadwal Produksi dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.20 Implementasi Halaman Detail Jadwal Induk Produksi



Sablon Malang App

Ubah Jadwal Produksi

Forms

Tanggal Produksi

07/08/2018

Tanggal Selesai

07/07/2018

MPS

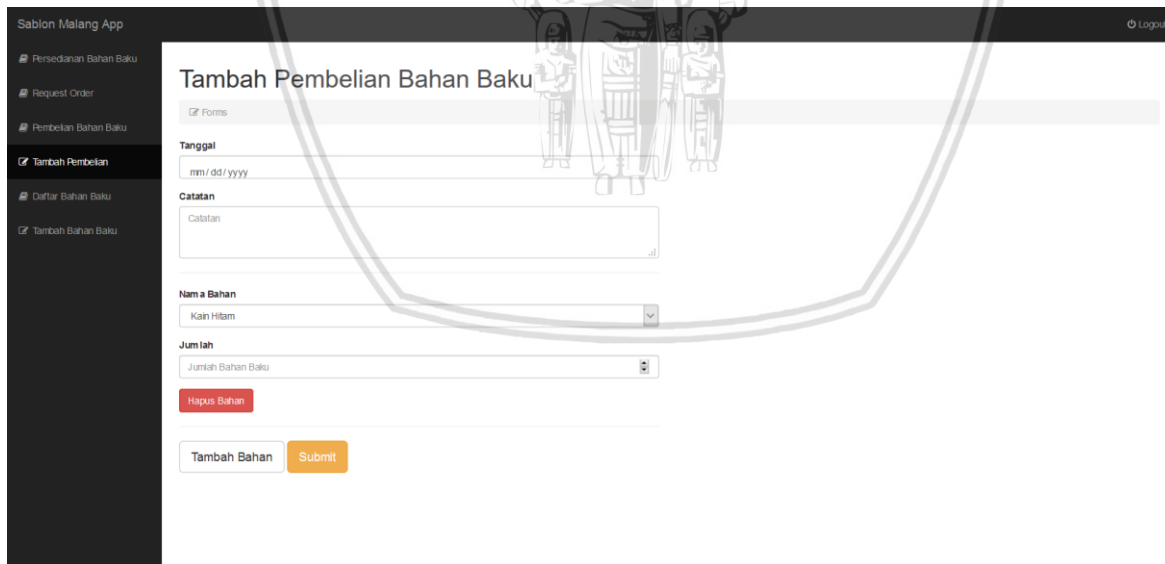
40

Submit

Gambar 5.54 Implementasi Halaman Detail Jadwal Produksi

Gambar 5.54 merupakan tampilan detail Jadwal Induk Produksi berdasarkan Jadwal Induk Produksi yang sudah dipilih dan bisa diubah. Halaman ini bisa diakses ketika memilih tombol Edit yang berada pada daftar Jadwal Produksi. Tampilan antarmuka halaman Detail Jadwal Produksi dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.21 Implementasi Halaman Tambah Pembelian



Sablon Malang App

Tambah Pembelian Bahan Baku

Forms

Tanggal

mm/dd/yyyy

Catatan

Catatan

Nama Bahan

Kain Hitam

Jumlah

Jumlah Bahan Baku

Hapus Bahan

Tambah Bahan Submit

Gambar 5.55 Implementasi Halaman Tambah Pembelian

Gambar 5.55 merupakan tampilan form Tambah Pembelian. Tampilan antarmuka halaman Tambah Pembelian dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.22 Implementasi Halaman Daftar Pembelian

No	Nama Bahan	Tempat	Jumlah	Tanggal	Catatan	Option
1	Kain Hitam	konveksi	12	29 Jun 2018	Pembelian Sesuai Permintaan	Edit
2	Tinta Hijau	sablon	50	26 Jun 2018	-	Edit

Gambar 5.56 Implementasi Halaman Daftar Pembelian

Gambar 5.56 merupakan tampilan tabel dari semua Pembelian. Tampilan antarmuka halaman Daftar Pembelian dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.23 Implementasi Halaman Detail Status Pembelian

Detail Pembelian Bahan Baku

☐ Forms

Tanggal
06/29/2018

Catatan
Pembelian Sesuai Permintaan

Nama Bahan
Kain Hitam

Jumlah
12

[Submit](#)

Gambar 5.57 Implementasi Halaman Detail Pembelian

Gambar 5.57 merupakan tampilan detail Pembelian berdasarkan Pembelian yang sudah dipilih dan bisa diubah. Halaman ini bisa diakses ketika memilih tombol Edit yang berada pada daftar Pembelian. Tampilan antarmuka halaman Detail Pembelian dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.24 Implementasi Halaman Tambah User

Sablon Malang App

Logout

Daftar Pesanan
Jadwal Produksi
Catatan Pesanan
Persediaan Bahan Baku
Request Order
Pembelian Bahan Baku
Daftar Bahan Baku
Daftar User
Tambah User

User Baru

Forms

Nama User
Kasir

Username
Username

Password
password

Submit

Gambar 5.58 Implementasi Halaman Tambah User

Gambar 5.58 merupakan tampilan form Tambah User. Tampilan antarmuka halaman Tambah User dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.25 Implementasi Halaman Daftar User

Sablon Malang App

Logout

Daftar Pesanan
Jadwal Produksi
Catatan Pesanan
Persediaan Bahan Baku
Request Order
Pembelian Bahan Baku
Daftar Bahan Baku
Daftar User
Tambah User

Daftar User

Show 10 entries

Search:

No	Nama User	Username	Option
1	permesok	permesok	Edit Hapus
2	kasir	kasir	Edit Hapus
3	konveksi	konveksi	Edit Hapus
4	sablon	sablon	Edit Hapus
5	admin	admin	Edit Hapus
No	Nama User	Username	Option

Showing 1 to 5 of 5 entries

Previous 1 Next

Gambar 5.59 Implementasi Halaman Daftar User

Gambar 5.59 merupakan tampilan tabel dari semua data user. Tampilan antarmuka halaman Daftar User dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.26 Implementasi Halaman Detail User

Sablon Malang App

Detail User

Forms

Nama User

pemasok

Username

pemasok

Password Baru

Submit

Gambar 5.60 Implementasi Halaman Detail User

Gambar 5.60 merupakan tampilan detail User berdasarkan data User yang sudah dipilih dan bisa diubah. Halaman ini bisa diakses ketika memilih tombol Edit yang berada pada daftar User. Tampilan antarmuka halaman Detail User dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

5.2.6.27 Implementasi Halaman Nota

Nota Pesanan

Nama Pemesan	Fakhruzzahid	Harga Satuan	100000
Alamat	Malang	Total Harga	1200000
Jenis Pesanan	Konveksi dan Sablon	Pembayaran	DP (100000)
Banyak	12		
Tanggal Pesanan	01 Jul 2018	Malang, 01 Jul 2018	
Tanggal Selesai	24 Jul 2018		
Catatan	-	Kasir	

Gambar 5.61 Implementasi Halaman Nota

Gambar 5.61 merupakan tampilan Halaman Nota saat dicetak.. Tampilan antarmuka halaman Nota dibuat berdasarkan hasil dari perancangan antarmuka sebelumnya.

BAB 6 PENGUJIAN

Tahap Pengujian dilakukan setelah implementasi selesai. Pengujian dilakukan untuk memeriksa apakah hasil implementasi sesuai dengan rekayasa kebutuhan dan perancangan atau tidak. Pengujian yang akan dilakukan adalah pengujian unit dan pengujian validasi.

6.1 Pengujian Unit

Pengujian Unit digunakan untuk menguji setiap unit terkecil yang dihasilkan dari proses perancangan. Pengujian unit dilakukan dengan menggunakan *basis path testing* yang merupakan salah satu jenis dari pengujian *white box testing*. Pengujian ini akan dilakukan pada tiga sampel uji yaitu Method ProsesTambah pada klas Pesananbaru, Method prosesTambah pada klas Catatanbaru, dan method Method getBahan pada klas Bahan.

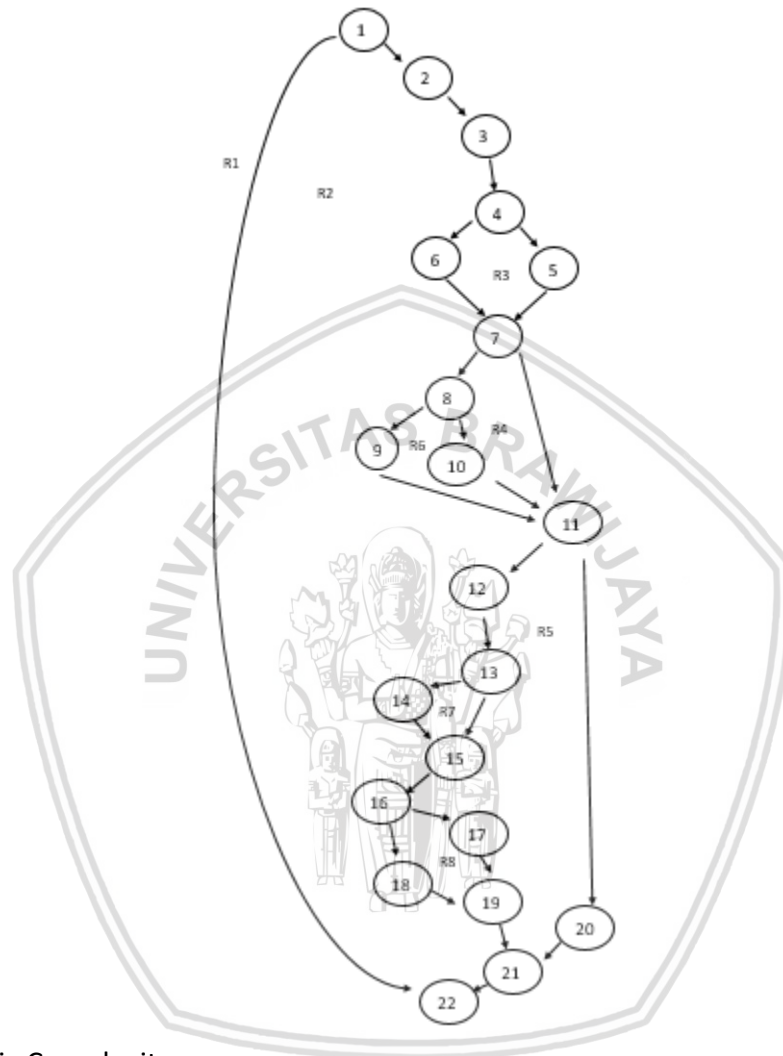
6.1.1 Pengujian Unit Method ProsesCek pada klas CatatanBaru

<i>If (post=inputan)</i>	1
<i>Instansiasi variable pesan = false</i> <i>Mengambil data persediaan</i> <i>Instansiasi variable array</i>	2
<i>Foreach (data persediaan berdasarkan baris)</i> <i>Data persediaan disimpan dalam variable array</i> <i>End foreach</i>	3
<i>Instansiasi variable array</i> <i>Foreach (jumlah inputan bahan)</i> <i>If (data bahan berhasil diinputkan)</i>	4
<i>Inisialisasi data array dengan data inputan</i> <i>End if</i>	5
<i>Else</i> <i>Memasukkan langsung data cek bahan</i> <i>End foreach</i>	6
<i>Foreach (data cek bahan berdasarkan jumlah inputan)</i> <i>If (data bahan berhasil di inputkan)</i>	7
<i>If (jumlah persediaan < inputan data bahan)</i>	8
<i>Menampilkan pesan "Jumlah Melebihi Kapasitas"</i>	9

<i>End if</i>	
<i>Else</i> <i>Menampilkan Pesan "Data Bahan Tidak ditemukan"</i>	10
<i>End if</i> <i>End foreach</i> <i>Jika (tidak menampilkan pesan error)</i>	11
<i>Inisialisasi variable IDPesanan</i> <i>Inisialisasi variable status</i> <i>Inisialisasi variable tempat</i> <i>Membuat variable untuk memanggil method tambah di database</i> <i>Membuat variable untuk memanggil method getDetail di database</i> <i>Instansiasi variable array</i>	12
<i>Foreach (hasil dari data inputan)</i> <i>If (inputan tempat = data tempat)</i>	13
<i>Inisialisasi variable array dengan data IDProduksiPesanan</i> <i>End if</i> <i>End foreach</i>	14
<i>Update produksi pesanan statuspesanan menjadi proses</i> <i>Instansiasi variable array</i>	15
<i>Foreach (jumlah inputan bahan)</i> <i>If (data bahan berhasil di inputkan)</i>	16
<i>Inisialisasi data array dengan data inputan</i> <i>End if</i>	17
<i>Else</i> <i>Memasukkan langsung data inputan</i> <i>End foreach</i>	18
<i>Memasukkan data catatan dengan data inputan</i> <i>Kembali ke halaman detail pesanan</i> <i>End if</i>	19
<i>Else</i> <i>Memanggil method cek</i>	20
<i>End if</i>	21

Basis Path Testing

a. Flow Graph



b. Cyclomatic Complexity

$$V(G) = \text{Jumlah Region} = 8$$

$$VG = E - N + 2 = 28 - 22 + 2 = 8$$

$$V(G) = P + 1 = 7 + 1 = 8$$

c. Independent Path

$$\text{Jalur 1} = 1 - 22$$

$$\text{Jalur 2} = 1 - 2 - 3 - 4 \dots 7 - 8 - 9 - 11 - 20 - 21 - 22$$

$$\text{Jalur 3} = 1 - 2 - 3 - 4 \dots 7 - 8 - 10 - 11 - 20 - 21 - 22$$

$$\text{Jalur 4} = 1 - 2 - 5 - 7 - 11 - 12 - 13 \dots 16 - 17 - 19 - 21 - 22$$

$$\text{Jalur 5} = 1 - 2 - 6 - 7 - 11 - 12 - 13 \dots 16 - 18 - 19 - 21 - 22$$

$$\text{Jalur 6} = 1 - 2 \dots 7 - 11 - 2 - 12 - 13 - 14 - 15 \dots 19 - 21 - 22$$

$$\text{Jalur 7} = 1 - 2 \dots 7 - 11 - 2 - 12 - 13 - 15 \dots 19 - 21 - 22$$

$$\text{Jalur 8} = 1 - 2 - 3 - 4 \dots 7 - 11 - 20 - 21 - 22$$

Test case dan hasil akan dijelaskan pada Table 6.1 dibawah ini

Tabel 6.1 Hasil Pengujian Unit method prosesCheck pada klas CatatanBaru

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Post = null	Menampilkan kembali form Catatan Pesanan	Tampilan form Catatan Pesanan	Valid
2	2	Memasukkan Post = (NamaBahan = "Kain Hitam", Jumlah = "1000", Catatan = "-" dan Tanggal Catatan = "07/08/2019"). Mengirim pesan message error	Message eror = "Jumlah bahan Melebihi Kapasitas". Menampilkan kembali formCatatanPesanan "	Tampilan form catatan pesanan dengan menampilkan pesan eror "Jumlah bahan Melebihi Kapasitas"	Valid
3	3	Memasukkan Post = (NamaBahan = "Kain Pelangi", Jumlah = "0", Catatan = "-" dan Tanggal Catatan = "07/08/2019"). Mengirim pesan message error	Message error = "Data Bahan Tidak Ditemukan". Menampilkan kembali form Catatan Pesanan	Tampilan form catatan pesanan dengan menampilkan pesan eror "Data Bahan Tidak Ditemukan"	Valid
4	4	Memasukkan dua data dengan Post = (NamaBahan = "Kain Hitam", Jumlah = "10", Catatan = "-" dan Tanggal Catatan = "07/08/2019"), (NamaBahan = "Kain Hitam", Jumlah = "5", Catatan = "-" dan	Bahan yang input menjadi satu data dengan Jumlah Bahan = (IDBahan 1 + ID Bahan 2) menyimpan data post = (NamaBahan = "Kain Hitam", Jumlah = "10", Catatan = "-" dan Tanggal Catatan = "07/08/2019"), (NamaBahan = "Kain Hitam", Jumlah = "5",	Menampilkan halaman daftar pesanan dengan data yang sudah disimpan sementara yaitu NamaBahan = "Kain Hitam", Jumlah = "15", Catatan = "-" dan Tanggal Catatan = "07/08/2019" dan	Valid

		Tanggal Catatan = "07/08/2019"). Membuat variable dengan memanggil method tambah untuk menyimpan data sesuai masukkan. Masuk kedalam fungsi foreach dengan Memasukkan semua data post.	Catatan = "-" dan Tanggal Catatan = "07/08/2019"). Menampilkan kembali "daftarpesanan" dan mengubah statusPesanan = "statusproses"	Statuspesanan berubah menjadi = "statusproses"	
5	5	Memasukkan dua data dengan Post = (NamaBahan = "Kain Hitam", Jumlah = "10", Catatan = "-" dan Tanggal Catatan = "07/08/2019"), (NamaBahan = "Kain Hijau", Jumlah = "5", Catatan = "-" dan Tanggal Catatan = "07/08/2019"). Membuat variable dengan memanggil method tambah untuk menyimpan data sesuai masukkan. Masuk kedalam fungsi foreach dengan Memasukkan semua data post.	Terdapat 2 data bahan yang input dengan menyimpan data post = (NamaBahan = "Kain Hitam", Jumlah = "10", Catatan = "-" dan Tanggal Catatan = "07/08/2019"), (NamaBahan = "Kain Hijau", Jumlah = "5", Catatan = "-" dan Tanggal Catatan = "07/08/2019"). kemudian Menampilkan kembali "daftarpesanan" dengan data yang sudah disimpan sementara dan mengubah statusPesanan = "statusproses"	Menampilkan halaman daftar pesanan dengan data yang sudah disimpan sementara yaitu NamaBahan = "Kain Hitam", Jumlah = "15", Catatan = "-" dan Tanggal Catatan = "07/08/2019" dan NamaBahan = "Kain Hijau", Jumlah = "5", Catatan = "-" dan Tanggal Catatan = "07/08/2019" dan Statuspesanan berubah menjadi = "statusproses"	Valid
6	6	Tempat = "Konveksi" Memasukkan data dengan Post = (NamaBahan =	Mengubah statuspesanan dalam tempat = "konveksi" menjadi "proses"	Statusproses berhasil berubah menjadi "proses"	Valid

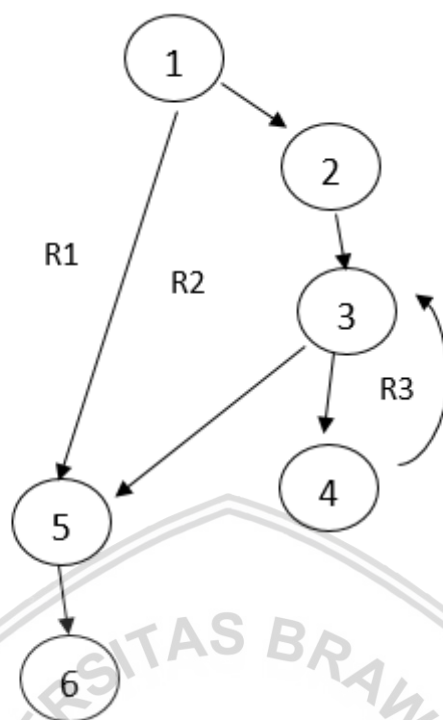
		"Kain Hitam", Jumlah = "10", Catatan = "-" dan Tanggal Catatan = "07/08/2019"			
7	7	Tempat = "sablon" Memasukkan data dengan Post = (NamaBahan = "Kain Hitam", Jumlah = "10", Catatan = "-" dan Tanggal Catatan = "07/08/2019"	Mengubah statuspesanan dalam tempat = "sablon" menjadi "proses"	Statusproses berhasil berubah menjadi "proses"	Valid
8	8	Pesan error = true	Menampilkan kembali form Catatan Pesanan	Tampilan form Catatan Pesanan	

6.1.2 Pengujian Unit Method prosesTambah pada klas RequestOrder

<i>If (post=inputan)</i>	1
<i>Inisialisasi variable id ke model requestorder</i> <i>membuat variable array</i>	2
<i>For (Berhenti ketika i < jumlah IDBahan)</i>	3
<i>Inisialisasi variable array berdasarkan</i> <i>IDRequestOrder</i> <i>IDBahan</i> <i>Jumlah</i> <i>End for</i>	4
<i>Menyimpan masukan ke dalam database requestorderbahan dengan variable</i> <i>array</i> <i>Kembali ke requestOrder</i> <i>End if</i>	5
<i>Memanggil method tambah</i>	6

Basis Path Testing

- Flow Graph



- Cyclomatic Complexity
 $V(G) = \text{Jumlah Region} = 3$
 $VG = E - N + 2 = 7 - 6 + 2 = 3$
 $V(G) = P + 1 = 2 + 1 = 3$
- Independent Path
 Jalur 1 = 1 – 7
 Jalur 2 = 1 – 2 – 3 – 4 – 3 – 5 – 6
 Jalur 3 = 1 – 2 – 3 – 4 – 3 – 4 – 3 – 5 – 6

Test case dan hasil akan dijelaskan pada Table 6.2 dibawah ini

Tabel 6.2 Hasil Pengujian Unit method prosesTambah pada klas RequestOrder

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Post = null	Menampilkan kembali form jadwal produksi baru	Tampilan form jadwal produksi baru	Valid
2	2	Memasukkan Post = (tanggal = "07/03/2018", catatan = "-", namaBahan =	Menyimpan data post = (IDRequestOrder = "1", tanggal = "07/03/2018", catatan = "-", namaBahan =	Tampilan daftar Request Order dengan satu data inputan berhasil ditambah yaitu (tanggal = "07/03/2018",	Valid

		<p>"Tinta Hijau", dan jumlah = "13").</p> <p>Membuat variable dengan memanggil method tambah untuk menyimpan data sesuai masukkan</p>	<p>"Tinta Hijau", dan jumlah = "13") dan mengembalikan halaman "requestorder"</p>	<p>catatan = "-", namaBahan = "Tinta Hijau", dan jumlah = "13")</p>	
3	3	<p>Memasukkan Post = (tanggal = "07/03/2018", catatan = "-", namaBahan = "Tinta Hijau", dan jumlah = "14")(tanggal = "07/03/2018", catatan = "-", namaBahan = "Cat Kuning", dan jumlah = "20)</p> <p>Membuat variable dengan memanggil method tambah untuk menyimpan data sesuai masukkan. Masuk kedalam fungsi for (count = 2). Memasukkan semua data post.</p>	<p>Menyimpan data post (IDRequestOrder = "2", tanggal = "07/03/2018", catatan = "-", namaBahan = "Tinta Hijau", dan jumlah = "14")(IDRequestOrder = "2", tanggal = "07/03/2018", catatan = "-", namaBahan = "Cat Kuning", dan jumlah = "20) dan mengembalikan halaman "requestorder"</p>	<p>Tampilan daftar Request Order dengan dua data inputan berhasil ditambah yaitu (tanggal = "07/03/2018", catatan = "-", namaBahan = "Tinta Hijau", dan jumlah = "14")(tanggal = "07/03/2018", catatan = "-", namaBahan = "Cat Kuning", dan jumlah = "20)</p>	Valid

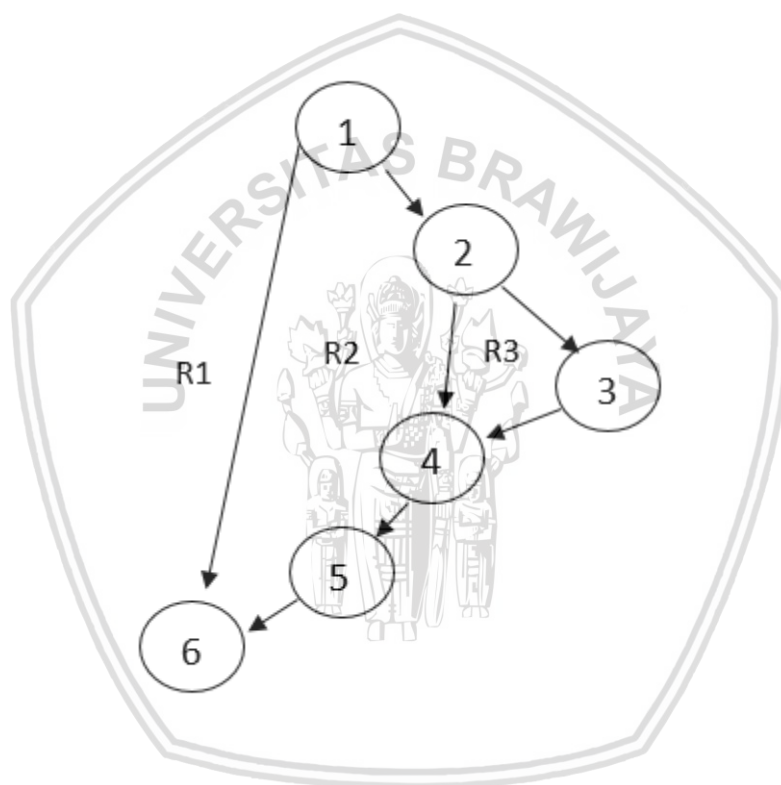
6.1.3 Pengujian Unit Method prosesTambah pada klas JadwalProduksiBaru

<i>If (post=inputan)</i>	1
<i>If (akses user = konveksi)</i>	2
<i>Memasukkan data jadwal produksi berdasarkan tempat konveksi</i>	3
<i>End if</i>	
<i>Else</i>	4

Memasukkan data jadwal produksi berdasarkan tempat sablon End else	
Memanggil method tambah dari database jadwal produksi berdasarkan inputan Kembali ke halaman jadwal produksi End if	5
Kembali ke halaman jadwal produksi	6

Basis Path Testing

b. Flow Graph



c. Cyclomatic Complexity

$$V(G) = \text{Jumlah Region} = 3$$

$$VG = E - N + 2 = 7 - 6 + 2 = 3$$

$$V(G) = P + 1 = 2 + 1 = 3$$

d. Independent Path

$$\text{Jalur 1} = 1 - 2 - 5 - 6$$

$$\text{Jalur 2} = 1 - 2 - 3 - 4 - 5 - 6$$

$$\text{Jalur 3} = 1 - 2 - 3 - 4 - 2 - 3 - 4 - 5 - 6$$

Test case dan hasil akan dijelaskan pada Table 6.3 dibawah ini

Tabel 6.3 Hasil Pengujian Unit method prosesTambah pada klas JadwalProduksiBaru

No	No Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Post = null	Menampilkan kembali form Request Order baru	Tampilan form Request Order baru	Valid
2	2	Protected_item = "konveksi" dan memasukkan Post = (jadwalProduksi = "07/11/2018", jadwalSelesai = "07/18/2018" dan MPS = "50"). Memanggil method tambah untuk menyimpan data sesuai masukkan	Menyimpan data post = (IDJadwalProduksi = "1", jadwalProduksi = "07/11/2018", jadwalSelesai = "07/18/2018", tempat = "konveksi" dan MPS = "50") dan mengembalikan halaman "jadwalproduksi"	Tampilan daftar Jadwal Produksi Konveksi dengan satu data inputan berhasil ditambah (jadwalProduksi = "07/11/2018", jadwalSelesai = "07/18/2018", dan MPS = "50")	Valid
3	3	Protected_item = "Sablon" dan memasukkan Post = (jadwalProduksi = "07/11/2018", jadwalSelesai = "07/18/2018" dan MPS = "30"). Memanggil method tambah untuk menyimpan data sesuai masukkan	Menyimpan data post = (IDJadwalProduksi = "2", jadwalProduksi = "07/11/2018", jadwalSelesai = "07/18/2018", tempat = "sablon" dan MPS = "50") dan mengembalikan halaman "jadwalproduksi"	Tampilan daftar Jadwal Produksi Sablon dengan satu data inputan berhasil ditambah (jadwalProduksi = "07/11/2018", jadwalSelesai = "07/18/2018", dan MPS = "50")	Valid

6.2 Pengujian Validasi

Pengujian Validasi digunakan untuk mengetahui apakah system yang telah dibangun sesuai dengan seluruh kebutuhan yang telah ditetapkan atau tidak. Pengujian validasi dilakukan dengan cara memeriksa apakah sistem sudah berjalan dengan baik dan tidak ada error yang terjadi. Proses pengujian validasi mengacu pada daftar kebutuhan fungsional maupun non-fungsional yang telah ditetapkan berdasarkan hasil proses rekayasa kebutuhan. Pengujian validasi akan dilakukan pada semua system dengan menggunakan metode scenario-based testing yang merupakan pengujian dari *black box testing*.

6.2.1 Pengujian Login

Pengujian validasi *login* dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0100. Berikut ini adalah prosedur kasus uji yang dilakuakn.

- a. Kasus Uji Berhasil Login Sebagai kasir
Kasus uji berhasil login sebagai kasir akan dijelaskan pada tabel 6.4

Tabel 6.4 Kasus Uji Berhasil Login Sebagai Kasir

Nama Kasus Uji	Kasus Uji Berhasil Login Sebagai Kasir
Prosedur	<ol style="list-style-type: none"> 1. Masuk kedalam Halaman Login 2. Memasukkan username = kasir dan password = malang 3. Menekan tombol login
Hasil yang diharapkan	Sistem akan menampilkan halaman awal kasir
Hasil	Sistem menampilkan halaman awal kasir
Status	Valid

- b. Kasus Uji Berhasil Login Sebagai Gudang
Kasus uji berhasil login sebagai Gudang akan dijelaskan pada tabel 6.5

Tabel 6.5 Kasus Uji Berhasil Login Sebagai Gudang

Nama Kasus Uji	Kasus Uji Berhasil Login Sebagai Gudang
Prosedur	<ol style="list-style-type: none"> 1. Masuk kedalam Halaman Login 2. Memasukkan username = Gudang dan password = malang 3. Menekan tombol login
Hasil yang diharapkan	Sistem akan menampilkan halaman awal Gudang
Hasil	Sistem menampilkan halaman awal Gudang
Status	Valid

- c. Kasus Uji Berhasil Login Sebagai Sablon
Kasus uji berhasil login sebagai Sablon akan dijelaskan pada tabel 6.6

Tabel 6.6 Kasus Uji Berhasil Login Sebagai Sablon

Nama Kasus Uji	Kasus Uji Berhasil Login Sebagai Sablon
Prosedur	<ol style="list-style-type: none"> 1. Masuk kedalam Halaman Login 2. Memasukkan username = sablon dan password = malang 3. Menekan tombol login
Hasil yang diharapkan	Sistem akan menampilkan halaman awal Sablon
Hasil	Sistem menampilkan halaman awal Sablon
Status	Valid

d. Kasus Uji Berhasil Login Sebagai Konveksi

Kasus uji berhasil login sebagai Konveksi akan dijelaskan pada tabel 6.7

Tabel 6.7 Kasus Uji Berhasil Login Sebagai Konveksi

Nama Kasus Uji	Kasus Uji Berhasil Login Sebagai Konveksi
Prosedur	<ol style="list-style-type: none"> 1. Masuk kedalam Halaman Login 2. Memasukkan username = konveksi dan password = malang 3. Menekan tombol login
Hasil yang diharapkan	Sistem akan menampilkan halaman awal Konveksi
Hasil	Sistem menampilkan halaman awal Konveksi
Status	Valid

e. Kasus Uji Data Login Kosong

Kasus uji Data Login Kosong akan dijelaskan pada tabel 6.8

Tabel 6.8 Kasus Uji Data Login Kosong

Nama Kasus Uji	Kasus Uji Data Login Kosong
Prosedur	<ol style="list-style-type: none"> 1. Masuk kedalam Halaman Login 2. Tidak Memasukkan username dan password 3. Menekan tombol login
Hasil yang diharapkan	Sistem akan menampilkan pesan username dan password salah
Hasil	Sistem menampilkan pesan username dan password salah
Status	Valid

f. Kasus Uji Data Login Salah

Kasus uji Data Login Salah akan dijelaskan pada tabel 6.9

Tabel 6.9 Kasus Uji Data Login Salah

Nama Kasus Uji	Kasus Uji Data Login Salah
Prosedur	<ol style="list-style-type: none"> 1. Masuk kedalam Halaman Login 2. Memasukkan username = Gudang dan password = surabaya 3. Menekan tombol login
Hasil yang diharapkan	Sistem akan menampilkan pesan username dan password salah
Hasil	Sistem menampilkan pesan username dan password salah
Status	Valid

6.2.2 Pengujian Logout

Pengujian validasi *logout* dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0200. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji logout akan dijelaskan pada tabel 6.10

Tabel 6.10 Kasus Uji Logout

Nama Kasus Uji	Kasus Uji Berhasil User berhasil logout
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem 2. User menekan tombol logout
Hasil yang diharapkan	Sistem akan menampilkan halaman login
Hasil	Sistem menampilkan halaman login
Status	Valid

6.2.3 Pengujian Membuat Pesanan

Pengujian validasi Berhasil Membuat Pesanan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0300. . Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Membuat Pesanan

Kasus uji Berhasil Membuat Pesanan akan dijelaskan pada tabel 6.11

Tabel 6.11 Kasus Uji Berhasil Membuat Pesanan

Nama Kasus Uji	Kasus Uji Berhasil Membuat Pesanan
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Kasir 2. Kasir memilih menu Tambah Pesanan 3. Sistem menampilkan form pesanan baru

	<p>4. Kasir mengisi form pesanan baru dengan isi Nama Pemesan = "Fakhruzzahid", Nomor HP "082141270696", Alamat = "Malang", Jenis Pesanan "Konveksi dan Sablon", Banyak = "12", Harga Satuan "100000", Total Harga = "1200000", Pembayaran = "100000", Tanggal Pesanan "07/01/2018", Tanggal Selesai = "07/24/2018", Catatan= "-", Gambar = "Clawsea.png".</p> <p>5. Kasir memilih Jadwal Produksi yang berisi Jadwal Produksi Konveksi dengan Tanggal Produksi = "07/08/2018" dan Jadwal Produksi Sablon dengan Tanggal produksi = "07/11/2018"</p> <p>6. Kasir menekan tombol submit</p>
Hasil yang diharapkan	Sistem akan menyimpan data pesanan kedalam database dan menampilkan kembali ke daftar pesanan
Hasil	Data pesanan telah disimpan dan menampilkan halaman daftar pesanan yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form Pesanan Kosong
Kasus uji Form Pesanan Kosong akan dijelaskan pada tabel 6.12

Tabel 6.12 Kasus Uji Form Pesanan Kosong

Nama Kasus Uji	Kasus Uji Form Pesanan Kosong
Prosedur	<p>1. User masuk sebagai Kasir</p> <p>2. Kasir memilih menu Tambah Pesanan</p> <p>3. Sistem menampilkan form pesanan baru</p> <p>4. Kasir tidak mengisi Nama Pemesan</p> <p>5. Kasir menekan tombol submit</p>
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form pesanan tidak boleh kosong
Hasil	System menampilkan pesan form pesanan tidak boleh kosong
Status	Valid

- c. Kasus uji Form Pesanan Jumlah Pesanan Melebihi Kapasitas Produksi
Kasus uji Form Pesanan Jumlah Pesanan Melebihi Kapasitas Produksi akan dijelaskan pada tabel 6.13.

Tabel 6.13 Kasus Uji Form Jumlah Pesanan Melebihi Kapasitas Produksi

Nama Kasus Uji	Kasus Uji Form Pesanan Pesanan Melebihi Kapasitas Produksi
-----------------------	--

Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Kasir 2. Kasir memilih menu Tambah Pesanan 3. Sistem menampilkan form pesanan baru 4. Kasir mengisi form pesanan dengan data Nama Pemesan = "Fakhruzzahid", Nomor HP "082141270696", Alamat = "Malang", Jenis Pesanan "Konveksi dan Sablon", produk="Jaket Hitam Sablon Rubber", Banyak = "50", Harga Satuan "100000", Total Harga = "1200000", Pembayaran = "100000", Tanggal Pesanan "07/01/2018", Tanggal Selesai = "07/24/2018", Catatan= "-", Gambar = "Clawsea.png". 5. Kasir memilih Jadwal Produksi yang berisi Jadwal Produksi Konveksi dengan Tanggal Produksi = "07/08/2018" dan Jadwal Produksi Sablon dengan Tanggal produksi = "07/11/2018" 6. Kasir menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa pesanan melebihi kapasitas
Hasil	System menampilkan Jadwal Produksi melebihi kapasitas
Status	Valid

6.2.4 Pengujian Melihat Pesanan

Pengujian validasi Melihat Pesanan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0400. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Pesanan akan dijelaskan pada tabel 6.14.

Tabel 6.14 Kasus Uji Melihat Pesanan

Nama Kasus Uji	Kasus Uji Melihat Pesanan
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Kasir 2. Sablon memilih menu Daftar pesanan
Hasil yang diharapkan	Sistem akan menampilkan semua daftar pesanan yang berada pada database
Hasil	Sistem menampilkan semua daftar pesanan yang berada pada database
Status	Valid

6.2.5 Pengujian Berhasil Menghapus Pesanan

Pengujian validasi Berhasil Menghapus Pesanan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0500. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Berhasil Menghapus Pesanan akan dijelaskan pada tabel 6.15.

Tabel 6.15 Kasus Uji Berhasil Menghapus Pesanan

Nama Kasus Uji	Kasus Uji Berhasil Menghapus Pesanan
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Admin 2. System menampilkan halaman daftar pesanan 3. Admin menekan tombol hapus pada pesanan dengan nama pemesan = "Fakhruzzahid" 4. Sistem memberi pesan peringatan 5. Kasir menekan tombol ok
Hasil yang diharapkan	Sistem akan menghapus pesanan dari database
Hasil	Sistem menghapus pesanan dari database
Status	Valid

6.2.6 Pengujian Mengubah Pesanan

Pengujian validasi Mengubah Pesanan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0600. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Mengubah Pesanan
Kasus uji Berhasil Mengubah Pesanan akan dijelaskan pada tabel 6.16.

Tabel 6.16 Kasus Uji Berhasil Mengubah Pesanan

Nama Kasus Uji	Kasus Uji Berhasil Menambahkan Pesanan
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Kasir 2. Kasir memilih menu Daftar Pesanan 3. Sistem menampilkan Semua Daftar Pesanan 4. Kasir memilih salah satu pesanan yaitu dengan Nama Pemesan = "Fakhruzzahid" 5. Kasir mengisi Mengubah salah satu data yaitu Nama Pemesan = "Leo" 6. Kasir menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data pesanan terbaru kedalam database dan menampilkan kembali ke daftar pesanan
Hasil	Data pesanan terbaru telah disimpan dan menampilkan halaman daftar pesanan yang sudah diperbaharui

Status	Valid
---------------	-------

b. Kasus uji Form Pesanan Kosong

Kasus uji Form Pesanan Kosong akan dijelaskan pada tabel 6.17.

Tabel 6.17 Kasus Uji Form Pesanan Kosong

Nama Kasus Uji	Kasus Uji Form Pesanan Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Kasir 2. Kasir memilih menu Daftar Pesanan 3. Sistem menampilkan Semua Daftar Pesanan 4. Kasir memilih salah satu pesanan yaitu dengan Nama Pemesan = "Fakhruzzahid" 5. Kasir menghapus nama pemesan dan menjadi kosong 6. Kasir menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form pesanan tidak boleh kosong
Hasil	System menampilkan pesan form pesan tidak boleh kosong
Status	Valid

6.2.7 Pengujian Mengunggah Gambar Pesanan

Pengujian validasi Mengunggah Gambar Pesanan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0700. Berikut ini adalah prosedur kasus uji yang dilakukan.

a. Kasus uji Berhasil Mengunggah Gambar Pesanan

Kasus uji Berhasil Mengunggah Gambar Pesanan akan dijelaskan pada tabel 6.18.

Tabel 6.18 Kasus Uji Berhasil Mengunggah Gambar Pesanan

Nama Kasus Uji	Kasus Uji Berhasil Mengunggah Gambar Pesanan
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Kasir 2. Kasir memilih menu Tambah Pesanan 3. Sistem menampilkan form pesanan baru 4. Kasir Mengisi data form Pesanan dengan data Nama Pemesan = "Fakhruzzahid", Nomor HP "082141270696", Alamat = "Malang", Jenis Pesanan "Konveksi dan Sablon", produk="Jaket Hitam Sablon Rubber", Banyak = "12", Harga

	<p>Satuan “100000”, Total Harga = “1200000”, Pembayaran = “100000”, Tanggal Pesanan “07/01/2018”, Tanggal Selesai = “07/24/2018”, Catatan= “-”.</p> <p>5. Kasir memilih Jadwal Produksi yang berisi Jadwal Produksi Konveksi dengan Tanggal Produksi = “07/08/2018” dan Jadwal Produksi Sablon dengan Tanggal produksi = “07/11/2018”</p> <p>6. Kasir menekan tombol upload gambar pada form pesanan baru</p> <p>7. Sistem menampilkan folder tempat penyimpanan</p> <p>8. Kasir memilih Clawsea.png</p> <p>9. Kasir menekan tombol submit</p>
Hasil yang diharapkan	Sistem akan menyimpan file pesanan kedalam database dan menampilkan kembali ke daftar pesanan
Hasil	File pesanan telah disimpan dan menampilkan halaman daftar pesanan yang sudah diperbaharui
Status	Valid

- b. Kasus uji Mengunggah Gambar tidak Sesuai Format
 Kasus uji Mengunggah Gambar tidak Sesuai Format akan dijelaskan pada tabel 6.19.

Tabel 6.19 Kasus Uji Mengunggah Gambar tidak Sesuai Format

Nama Kasus Uji	Kasus Uji Mengunggah Gambar tidak Sesuai Format
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Kasir 2. Kasir memilih menu Tambah Pesanan 3. Sistem menampilkan form pesanan baru 4. Kasir Mengisi data form Pesanan dengan data Nama Pemesan = “Fakhruzzahid”, Nomor HP “082141270696”, Alamat = “Malang”, Jenis Pesanan “Konveksi dan Sablon”, produk=“Jaket Hitam Sablon Rubber”, Banyak = “12”, Harga Satuan “100000”, Total Harga = “1200000”, Pembayaran = “100000”, Tanggal Pesanan “07/01/2018”, Tanggal Selesai = “07/24/2018”, Catatan= “-”. 5. Kasir memilih Jadwal Produksi yang berisi Jadwal Produksi Konveksi dengan Tanggal Produksi = “07/08/2018” dan Jadwal Produksi Sablon dengan Tanggal produksi = “07/11/2018”

	6. Kasir menekan tombol upload gambar pada form pesanan baru 7. Sistem menampilkan folder tempat penyimpanan 8. Kasir memilih coba.txt 9. Kasir menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan gambar tidak sesuai format
Hasil	Sistem menampilkan pesan gambar tidak sesuai format
Status	Valid

6.2.8 Pengujian Melihat Bahan Konveksi

Pengujian validasi Melihat Bahan Konveksi dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0800. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Bahan Konveksi akan dijelaskan pada tabel 6.420.

Tabel 6.20 Kasus Uji Melihat Bahan Konveksi

Nama Kasus Uji	Kasus Uji Melihat Bahan Konveksi
Prosedur	1. User masuk kedalam sistem sebagai konveksi 2. Sistem menampilkan halaman utama konveksi 3. Konveksi memilih menu daftar persediaan
Hasil yang diharapkan	Sistem akan menampilkan semua daftar Bahan Baku yang berada pada konveksi
Hasil	Sistem menampilkan semua daftar Bahan Baku yang berada pada konveksi
Status	Valid

6.2.9 Pengujian Melihat Bahan Sablon

Pengujian validasi Melihat Bahan Sablon dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_0900. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Bahan Sablon akan dijelaskan pada tabel 6.21

Tabel 6.21 Kasus Uji Melihat Bahan Sablon

Nama Kasus Uji	Kasus Uji Melihat Bahan Sablon
Prosedur	1. User masuk kedalam sistem sebagai Sablon 2. Sistem menampilkan halaman utama Sablon 3. Sablon memilih menu daftar persediaan

Hasil yang diharapkan	Sistem akan menampilkan semua daftar Bahan Baku yang berada pada Sablon
Hasil	Sistem menampilkan semua daftar Bahan Baku yang berada pada Sablon
Status	Valid

6.2.10 Pengujian Menambahkan Bahan Baku

Pengujian validasi Menambahkan Bahan Baku dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1000. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Menambahkan Bahan
Kasus uji Berhasil Menambahkan Bahan akan dijelaskan pada tabel 6.22.

Tabel 6.22 Kasus Uji Berhasil Menambahkan Bahan Baku

Nama Kasus Uji	Kasus Uji Berhasil Menambahkan Bahan Baku
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Tambah Bahan Baku 3. Sistem menampilkan form bahan baku baru 4. Gudang mengisi form bahan baku baru dengan data Nama Bahan = "Kain Hitam", Tempat "Konveksi" dan Gambar = "Clawsea.png". 5. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data bahan baku kedalam database dan menampilkan kembali ke daftar bahan baku
Hasil	Data bahan baku telah disimpan dan menampilkan halaman daftar bahan baku yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form Bahan Baku Baru Kosong
Kasus uji Form Bahan Baku Baru Kosong akan dijelaskan pada tabel 6.23.

Tabel 6.23 Kasus Uji Form Bahan Baku Baru Kosong

Nama Kasus Uji	Kasus Uji Form Bahan Baku Baru Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Tambah Bahan Baku 3. Sistem menampilkan form bahan baku baru 4. Gudang tidak mengisi Nama Bahan 5. Gudang menekan tombol submit

Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form bahan baku baru tidak boleh kosong
Hasil	System menampilkan pesan form bahan baku baru tidak boleh kosong
Status	Valid

6.2.11 Pengujian Melihat Bahan Konveksi dan Sablon

Pengujian validasi Melihat Bahan Konveksi dan Sablon dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1100. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Bahan Konveksi dan Sablon akan dijelaskan pada tabel 6.23.

Tabel 6.23 Kasus Uji Melihat Bahan Konveksi dan Sablon

Nama Kasus Uji	Kasus Uji Melihat Bahan Konveksi dan Sablon
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Gudang 2. Sistem menampilkan halaman utama Gudang 3. Gudang memilih menu daftar Bahan Baku
Hasil yang diharapkan	Sistem akan menampilkan semua daftar Bahan Baku yang berada pada Konveksi dan Sablon
Hasil	Sistem menampilkan semua daftar Bahan Baku yang berada pada Konveksi dan Sablon
Status	Valid

6.2.12 Pengujian Menghapus Bahan Konveksi dan Sablon

Pengujian validasi Berhasil Menghapus Bahan Konveksi dan Sablon dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1200. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Berhasil Menghapus Bahan Konveksi dan Sablon akan dijelaskan pada tabel 6.24.

Tabel 6.24 Kasus Uji Berhasil Menghapus Bahan Konveksi dan Sablon

Nama Kasus Uji	Kasus Uji Berhasil Menghapus Bahan Konveksi dan Sablon
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Gudang 2. Sistem menampilkan halaman daftar Bahan Baku 3. Gudang menekan tombol hapus pada salah satu daftar Bahan Baku yaitu dengan Nama Bahan = "Kain Hitam". 4. Sistem memberi pesan peringatan 5. Gudang menekan tombol ok
Hasil yang diharapkan	Sistem akan menghapus Bahan Baku dari database

Hasil	Sistem menghapus Bahan Baku dari database
Status	Valid

6.2.13 Pengujian Mengubah Bahan Konveksi dan Sablon

Pengujian validasi Mengubah Bahan Konveksi dan Sablon dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1300. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Mengubah Bahan Konveksi dan Sablon
Kasus uji Berhasil Mengubah Bahan Konveksi dan Sablon akan dijelaskan pada tabel 6.25.

Tabel 6.25 Kasus Uji Berhasil Mengubah Bahan Konveksi dan Sablon

Nama Kasus Uji	Kasus Uji Berhasil Mengubah Bahan Konveksi dan Sablon
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih daftar bahan baku 3. Sistem menampilkan daftar bahan baku 4. Gudang memilih salah satu bahan baku dengan Nama Bahan = "Kain Hitam" kemudian menekan tombol edit 5. Sistem menampilkan detail bahan baku yang dipilih 6. Gudang mengganti data form bahan baku Nama Bahan = "Kain Putih". 7. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data bahan baku terbaru kedalam database dan menampilkan kembali ke daftar bahan baku
Hasil	Data bahan baku terbaru telah disimpan dan menampilkan halaman daftar bahan baku yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form Bahan Baku Kosong
Kasus uji Form Bahan Baku Kosong akan dijelaskan pada tabel 6.26.

Tabel 6.26 Kasus Uji Form Bahan Baku Kosong

Nama Kasus Uji	Kasus Uji Form Bahan Baku Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Sablon 2. Gudang memilih daftar bahan baku

	<ol style="list-style-type: none"> 3. Sistem menampilkan daftar bahan baku 4. Gudang memilih salah satu bahan baku dengan Nama Bahan "Kain Hitam" kemudian menekan tombol edit 5. Sistem menampilkan detail bahan baku yang dipilih 6. Gudang mengganti Nama Bahan menjadi kosong 7. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form bahan baku tidak boleh kosong
Hasil	System menampilkan pesan form bahan baku tidak boleh kosong
Status	Valid

6.2.14 Pengujian Mengunggah Gambar Bahan Baku

Pengujian validasi Mengunggah Gambar Bahan Baku dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1400. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Mengunggah Gambar Bahan Baku
Kasus uji Berhasil Mengunggah Gambar Bahan Baku akan dijelaskan pada tabel 6.27.

Tabel 6.27 Kasus Uji Berhasil Mengunggah Gambar Bahan Baku

Nama Kasus Uji	Kasus Uji Berhasil Mengunggah Gambar Bahan Baku
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Tambah Bahan Baku 3. Sistem menampilkan form bahan baku baru 4. Gudang mengisi form bahan baku baru dengan data Nama Bahan = "Kain Hitam", Tempat "Konveksi" 5. Gudang menekan tombol upload 6. Gudang memilih file Clawsea.png 7. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan file Bahan Baku kedalam database dan menampilkan kembali ke daftar Bahan Baku
Hasil	File Bahan Baku telah disimpan dan menampilkan halaman daftar Bahan Baku yang sudah diperbaharui

Status	Valid
--------	-------

- b. Kasus uji Mengunggah Gambar tidak Sesuai Format
Kasus uji Mengunggah Gambar tidak Sesuai Format akan dijelaskan pada tabel 6.28.

Tabel 6.28 Kasus Uji Mengunggah Gambar tidak Sesuai Format

Nama Kasus Uji	Kasus Uji Mengunggah Gambar tidak Sesuai Format
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Tambah Bahan Baku 3. Sistem menampilkan form bahan baku baru 4. Gudang mengisi form bahan baku baru dengan data Nama Bahan = "Kain Hitam", Tempat "Konveksi" 5. Gudang menekan tombol upload 6. Gudang memilih file coba.txt 7. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan gambar tidak sesuai format
Hasil	Sistem menampilkan pesan gambar tidak sesuai format
Status	Valid

6.2.15 Pengujian Membuat Pembelian Bahan Baku

Pengujian validasi Membuat Pembelian Bahan Baku dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1500. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Membuat Pembelian Bahan Baku
Kasus uji Berhasil Membuat Pembelian Bahan Baku akan dijelaskan pada tabel 6.29.

Tabel 6.29 Kasus Uji Berhasil Membuat Pembelian Bahan Baku

Nama Kasus Uji	Kasus Uji Berhasil Membuat Pembelian Bahan Baku
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Tambah Pembelian 3. Sistem menampilkan form Pembelian 4. Gudang mengisi form Pembelian baru dengan data Nama Bahan = "Kain Hitam", Tempat "Konveksi", Jumlah = "12" dan Catatan = "-". 5. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data pembelian kedalam database dan menampilkan kembali ke daftar pembelian

Hasil	Data pembelian telah disimpan dan menampilkan halaman daftar pembelian yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form Pembelian Kosong
Kasus uji Form Pembelian Kosong akan dijelaskan pada tabel 6.30.

Tabel 6.30 Kasus Uji Form Pembelian Kosong

Nama Kasus Uji	Kasus Uji Form Pembelian Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Tambah Pembelian 3. Sistem menampilkan form Pembelian 4. Gudang mengkosongkan data inputan Jumlah 5. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form pembelian tidak boleh kosong
Hasil	System menampilkan pesan form Pembelian baru tidak boleh kosong
Status	Valid

6.2.16 Pengujian Mengubah Pembelian Bahan Baku

Pengujian validasi Mengubah Pembelian Bahan baku dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1600. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Mengubah Pembelian Bahan Baku
Kasus uji Berhasil Mengubah Pembelian Bahan Baku akan dijelaskan pada tabel 6.31.

Tabel 6.31 Kasus Uji Berhasil Mengubah Pembelian Bahan Baku

Nama Kasus Uji	Kasus Uji Berhasil Mengubah Pembelian Bahan Baku
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih daftar pembelian 3. Sistem menampilkan daftar pembelian 4. Gudang memilih salah satu pembeliab dengan Nama Bahan = "Kain Hitam" kemudian menekan tombol edit 5. Sistem menampilkan detail pembelian yang dipilih 6. Gudang mengganti data form bahan baku Jumlah Bahan = "15". 7. Gudang menekan tombol submit

Hasil yang diharapkan	Sistem akan menyimpan data pembelian terbaru kedalam database dan menampilkan kembali ke daftar pembelian
Hasil	Data pembelian terbaru telah disimpan dan menampilkan halaman daftar pembelian yang sudah diperbaharui
Status	Valid

b. Kasus uji Form Pembelian Kosong

Kasus uji Form Pembelian Kosong akan dijelaskan pada tabel 6.32.

Tabel 6.32 Kasus Uji Form Pembelian Kosong

Nama Kasus Uji	Kasus Uji Form Pembelian Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Sablon 2. Gudang memilih daftar pembelian 3. Sistem menampilkan daftar pembelian 4. Gudang memilih salah satu pembelian dengan Nama Bahan "Kain Hitam" kemudian menekan tombol edit 5. Sistem menampilkan detail bahan baku yang dipilih 6. Gudang menghapus Jumlah dengan inputan kosong 7. Gudang menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form pembelian tidak boleh kosong
Hasil	System menampilkan pesan form pembelian tidak boleh kosong
Status	Valid

6.2.17 Pengujian Melihat Pembelian Bahan Baku

Pengujian validasi Melihat Pembelian Bahan Baku dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1700. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Pembelian Bahan Baku akan dijelaskan pada tabel 6.33.

Tabel 6.33 Kasus Uji Melihat Pembelian Bahan Baku

Nama Kasus Uji	Kasus Uji Melihat Pembelian Bahan Baku
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Gudang 2. Sistem menampilkan halaman utama Gudang 3. Gudang memilih menu daftar pembelian

Hasil yang diharapkan	Sistem akan menampilkan semua daftar pembelian
Hasil	Sistem menampilkan semua daftar pembelian
Status	Valid

6.2.18 Pengujian Menghapus Pembelian Bahan Baku

Pengujian validasi Berhasil Menghapus Pembelian Bahan Baku dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1800. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Berhasil Menghapus Pembelian Bahan Baku akan dijelaskan pada tabel 6.34.

Tabel 6.34 Kasus Uji Berhasil Menghapus Pembelian Bahan Baku

Nama Kasus Uji	Kasus Uji Berhasil Menghapus Pembelian Bahan Baku
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Gudang 2. Sistem menampilkan halaman daftar pembelian 3. Gudang menekan tombol hapus pada salah satu daftar pembelian yaitu dengan Nama Bahan = "Kain Hitam". 4. Sistem memberi pesan peringatan 5. Gudang menekan tombol ok
Hasil yang diharapkan	Sistem akan menghapus pembelian dari database
Hasil	Sistem menghapus pembelian dari database
Status	Valid

6.2.19 Pengujian Membuat Jadwal Induk Produksi

Pengujian validasi Membuat Jadwal Induk Produksi dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1900. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Membuat Jadwal Induk Produksi
Kasus uji Berhasil Membuat Jadwal Induk Produksi akan dijelaskan pada tabel 6.35.

Tabel 6.35 Kasus Uji Berhasil Membuat Jadwal Induk Produksi

Nama Kasus Uji	Kasus Uji Berhasil Membuat Jadwal Induk Produksi
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai sablon 2. Sablon memilih menu Tambah Jadwal Produksi 3. Sistem menampilkan form Jadwal Produksi 4. Sablon mengisi form Jadwal Produksi dengan data Jadwal Produksi = "07/11/2018", Jadwal Selesai = "07/18/2018" dan MPS = "50". 5. Sablon menekan tombol submit

Hasil yang diharapkan	Sistem akan menyimpan data jadwal produksi kedalam database dan menampilkan kembali ke daftar jadwal induk produksi
Hasil	Data jadwal produksi telah disimpan dan menampilkan halaman daftar jadwal induk produksi yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form Jadwal Produksi Baru Kosong
Kasus uji Form Jadwal Produksi Baru Kosong akan dijelaskan pada tabel 6.36.

Tabel 6.36 Kasus Uji Form Jadwal Produksi Baru Kosong

Nama Kasus Uji	Kasus Uji Form Jadwal Produksi Baru Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai sablon 2. Sablon memilih menu Tambah Jadwal Produksi 3. Sistem menampilkan form Jadwal Produksi 4. Sablon tidak mengisi inputan Tanggal Produksi 5. Sablon menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form jadwal produksi baru tidak boleh kosong
Hasil	Sistem menampilkan pesan form jadwal produksi baru tidak boleh kosong
Status	Valid

6.2.20 Pengujian Melihat Jadwal Induk Produksi

Pengujian validasi Melihat Jadwal Induk Produksi dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_1000. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Jadwal Induk Produksi akan dijelaskan pada tabel 6.37.

Tabel 6.37 Kasus Uji Melihat Jadwal Induk Produksi

Nama Kasus Uji	Kasus Uji Melihat Jadwal Induk Produksi
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai sablon 2. Sistem menampilkan halaman utama sablon 3. Sablon memilih menu Jadwal Produksi
Hasil yang diharapkan	Sistem akan menampilkan semua daftar Jadwal Induk Produksi yang telah diinputkan dan perhitungan yang sesuai
Hasil	Sistem menampilkan semua daftar Jadwal Induk Produksi yang telah diinputkan dan perhitungan yang sesuai

Status	Valid
--------	-------

6.2.21 Pengujian Mengubah Jadwal Induk Produksi

Pengujian validasi Mengubah Jadwal Induk Produksi dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2100. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Mengubah Jadwal Induk Produksi
Kasus uji Berhasil Mengubah Jadwal Induk Produksi akan dijelaskan pada tabel 38.

Tabel 6.38 Kasus Uji Berhasil Mengubah Jadwal Induk Produksi

Nama Kasus Uji	Kasus Uji Berhasil Mengubah Jadwal Induk Produksi
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Sablon 2. Sablon memilih menu Jadwal Produksi 3. Sistem menampilkan Daftar Jadwal Induk Produksi 4. Sablon memilih salah satu Jadwal dengan menekan tombol edit 5. Sistem menampilkan form jadwal produksi yang dipilih 6. Sablon mengganti data form jadwal produksi yang dengan data Jadwal Produksi = "07/11/2018", Jadwal Selesai = "07/18/2018" dan MPS = "60". 7. Sablon menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data Jadwal Produksi terbaru kedalam database dan menampilkan kembali ke daftar Jadwal Induk Produksi
Hasil	Data jadwal produksi terbaru telah disimpan dan menampilkan halaman daftar jadwal induk produksi yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form Jadwal Produksi Kosong
Kasus uji Form Jadwal Produksi Kosong akan dijelaskan pada tabel 6.39.

Tabel 6.39 Kasus Uji Form Jadwal Produksi Kosong

Nama Kasus Uji	Kasus Uji Form Jadwal Produksi Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Sablon 2. Sablon memilih menu Jadwal Produksi 3. Sistem menampilkan Daftar Jadwal Induk Produksi 4. Sablon memilih salah satu Jadwal dengan menekan tombol edit

	5. Sistem menampilkan form jadwal produksi yang dipilih 6. Sablon menghapus inputan MPS dengan inputan kosong 7. Sablon menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form jadwal produksi tidak boleh kosong
Hasil	System menampilkan pesan form jadwal produksi tidak boleh kosong
Status	Valid

6.2.22 Pengujian Menghapus Jadwal Induk Produksi

Pengujian validasi Berhasil Menghapus Jadwal Induk Produksi Sablon dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2200. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Berhasil Menghapus Jadwal Induk Produksi akan dijelaskan pada tabel 6.40.

Tabel 6.40 Kasus Uji Berhasil Menghapus Jadwal Induk Produksi

Nama Kasus Uji	Kasus Uji Berhasil Menghapus Jadwal Induk Produksi
Prosedur	1. User masuk kedalam sistem sebagai sablon 2. Sistem menampilkan halaman Jadwal Induk Produksi 3. Sablon menekan tombol hapus pada salah satu daftar Jadwal Induk Produksi 4. Sistem memberi pesan peringatan 5. Sablon menekan tombol ok
Hasil yang diharapkan	Sistem akan menghapus Jadwal Induk Produksi dari database
Hasil	Sistem menghapus Jadwal Induk Produksi dari database
Status	Valid

6.2.23 Pengujian Membuat Request Order

Pengujian validasi Membuat Request Order dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2300. Berikut ini adalah prosedur kasus uji yang dilakukan.

- Kasus uji Berhasil Membuat Request Order
Kasus uji Berhasil Membuat Request Order akan dijelaskan pada tabel 6.41.

Tabel 6.41 Kasus Uji Berhasil Membuat Request Order

Nama Kasus Uji	Kasus Uji Berhasil Membuat Request Order
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai sablon 2. Sablon memilih menu Tambah Request Order 3. Sistem menampilkan form Request Order 4. Sablon mengisi form Request Order dengan data Tanggal = "07/03/2018", Catatan = "-", Nama Bahan = "Tinta Hijau", dan Jumlah = "13". 5. Sablon menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data Request Order kedalam database dan menampilkan kembali ke daftar Request Order
Hasil	Data Request Order telah disimpan dan menampilkan halaman daftar Request Order yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form Request Order Kosong
 Kasus uji Form Request Order Kosong akan dijelaskan pada tabel 6.42.

Tabel 6.42 Kasus Uji Form Jadwal Produksi Baru Kosong

Nama Kasus Uji	Kasus Uji Form Request Order Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai sablon 2. Sablon memilih menu Tambah Request Order 3. Sistem menampilkan form Request Order 4. Sablon tidak mengisi inputan Nama Bahan 5. Sablon menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form Request Order baru tidak boleh kosong
Hasil	Sistem menampilkan pesan form Request Order baru tidak boleh kosong
Status	Valid

6.2.24 Pengujian Mengubah Request Order

Pengujian validasi Mengubah Request Order dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2500. Berikut ini adalah prosedur kasus uji yang dilakukan.

- c. Kasus uji Berhasil Mengubah Request Order
 Kasus uji Berhasil Mengubah Request Order akan dijelaskan pada tabel 43.

Tabel 6.43 Kasus Uji Berhasil Mengubah Request Order

Nama Kasus Uji	Kasus Uji Berhasil Mengubah Request Order
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Sablon 2. Sablon memilih menu Request Order 3. Sistem menampilkan Daftar Request Order 4. Sablon memilih salah satu Request Order dengan menekan tombol edit 5. Sistem menampilkan form Request Order yang dipilih 6. Sablon mengganti data form Request Order yang dengan data Tanggal = "07/03/2018", Catatan = "-", Nama Bahan = "Tinta Hijau", dan Jumlah = "23". 7. Sablon menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data Request Order terbaru kedalam database dan menampilkan kembali ke daftar Request Order
Hasil	Data jadwal produksi terbaru telah disimpan dan menampilkan halaman daftar Request Order yang sudah diperbaharui
Status	Valid

- d. Kasus uji Form Request Order Kosong
Kasus uji Form Request Order Kosong akan dijelaskan pada tabel 6.44.

Tabel 6.44 Kasus Uji Form Request Order Kosong

Nama Kasus Uji	Kasus Uji Form Request Order Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Sablon 2. Sablon memilih menu Request Order 3. Sistem menampilkan Daftar Request Order 4. Sablon memilih salah satu Jadwal dengan menekan tombol edit 5. Sistem menampilkan form Request Order yang dipilih 6. Sablon mengubah inputan Nama Bahan menjadi kosong 7. Sablon menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form Request Order tidak boleh kosong
Hasil	System menampilkan pesan form Request Order tidak boleh kosong
Status	Valid

6.2.25 Pengujian Melihat Request Order

Pengujian validasi Melihat Request Order dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2500. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Request Order akan dijelaskan pada tabel 6.45.

Tabel 6.45 Kasus Uji Melihat Request Order

Nama Kasus Uji	Kasus Uji Melihat Request Order
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai sablon 2. Sistem menampilkan halaman utama sablon 3. Sablon memilih menu Request Order
Hasil yang diharapkan	Sistem akan menampilkan semua daftar Request Order.
Hasil	Sistem menampilkan semua daftar Request Order yang telah diinputkan dan perhitungan yang sesuai
Status	Valid

6.2.26 Pengujian Menghapus Request Order

Pengujian validasi Berhasil Menghapus Request Order dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2600. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Berhasil Menghapus Request Order akan dijelaskan pada tabel 6.46.

Tabel 6.46 Kasus Uji Berhasil Menghapus Request Order

Nama Kasus Uji	Kasus Uji Berhasil Menghapus Request Order
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai sablon 2. Sistem menampilkan halaman Request Order 3. Sablon menekan tombol hapus pada data nama bahan = "Tinta Hijau" 4. Sistem memberi pesan peringatan 5. Sablon menekan tombol ok
Hasil yang diharapkan	Sistem akan menghapus Request Order dari database
Hasil	Sistem menghapus Request Order dari database
Status	Valid

6.2.27 Pengujian Memvalidasi Request Order

Pengujian validasi Memvalidasi Request Order dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2700. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Memvalidasi Disetujui Request Order

Kasus uji Memvalidasi Disetujui Request Order akan dijelaskan pada tabel 47.

Tabel 6.47 Kasus Uji Memvalidasi Disetujui Request Order

Nama Kasus Uji	Kasus Uji Memvalidasi Disetujui Request Order
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Request Order 3. Sistem menampilkan Daftar Request Order 4. Gudang memilih salah satu Request Order yaitu dengan Nama Bahan = "Tinta Hijau" 5. Sistem Menampilkan halaman Daftar Request Order
Hasil yang diharapkan	Sistem akan mengganti status Request Order menjadi disetujui
Hasil	Sistem mengganti status Request Order menjadi disetujui
Status	Valid

- b. Kasus uji Form Memvalidasi Tolak Request Order

Kasus uji Form Memvalidasi Tolak Request Order akan dijelaskan pada tabel 6.48.

Tabel 6.48 Kasus Uji Form Memvalidasi Tolak Request Order

Nama Kasus Uji	Kasus Uji Form Memvalidasi Tolak Request Order
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Gudang 2. Gudang memilih menu Request Order 3. Sistem menampilkan Daftar Request Order 4. Pemaosk memilih salah satu Request Order yaitu dengan Nama Bahan = "Tinta Kuning" 5. Gudang menekan tombol Tolak 6. Sistem Menampilkan halaman Daftar Request Order
Hasil yang diharapkan	Sistem akan mengganti status Request Order menjadi Tolak
Hasil	Sistem mengganti status Request Order menjadi Tolak
Status	Valid

6.2.28 Pengujian Mengecek Ketersediaan

Pengujian validasi Mengecek Ketersediaan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2800. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Mengecek Ketersediaan

Kasus uji Berhasil Mengecek Ketersediaan akan dijelaskan pada tabel 49.

Tabel 6.49 Kasus Uji Berhasil Mengecek Ketersediaan

Nama Kasus Uji	Kasus Uji Berhasil Mengecek Ketersediaan
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Konveksi 2. Konveksi memilih menu Jadwal Produksi 3. Sistem menampilkan Daftar Jadwal Produksi 4. Konveksi memilih salah satu Jadwal Produksi yaitu dengan tanggal produksi = "07/08/2018" dan menekan tombol pesanan 5. Sistem Menampilkan halaman List Pesanan 6. Konveksi memilih salah satu pesanan yang status pesanan "Diantrikan" yaitu dengan Nama Pemesan = "Fakhruzzahid" dan menekan tombol detail 7. Sistem menampilkan detail pesanan 8. Konveksi memilih tombol Check Ketersediaan 9. Sistem menampilkan form Check Ketersediaan 10. Konveksi mengisi form Check Ketersediaan dengan data Nama Bahan = "Kain Hitam", Jumlah = "4", Catatan = "-" dan Tanggal Catatan = "07/08/2019". 11. Konveksi Menekan Tombol Submit
Hasil yang diharapkan	Sistem akan menyimpan sementara data Check Ketersediaan dan bisa melanjutkan status pesanan ke status proses
Hasil	Sistem menyimpan sementara data Check Ketersediaan dan bisa melanjutkan status pesanan ke status proses
Status	Valid

b. Kasus uji Form Bahan Tidak Tersedia

Kasus uji Form Bahan Tidak Tersedia akan dijelaskan pada tabel 6.50.

Tabel 6.50 Kasus Uji Form Bahan Tidak Tersedia

Nama Kasus Uji	Kasus Uji Form Bahan Tidak Tersedia
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Konveksi 2. Konveksi memilih menu Jadwal Produksi 3. Sistem menampilkan Daftar Jadwal Produksi 4. Konveksi memilih salah satu Jadwal Produksi yaitu dengan tanggal produksi = "07/08/2018" dan menekan tombol pesanan 5. Sistem Menampilkan halaman List Pesanan 6. Konveksi memilih salah satu pesanan yang status pesanan "Diantrikan" yaitu dengan Nama

	Pemesan = "Fakhruzzahid" dan menekan tombol detail 7. Sistem menampilkan detail pesanan 8. Konveksi memilih tombol Check Ketersediaan 9. Sistem menampilkan form Check Ketersediaan 10. Konveksi mengisi form Check Ketersediaan dengan data Nama Bahan = "Kain Hitam", Jumlah = "100", Catatan = "-" dan Tanggal Catatan = "07/08/2019". 11. Konveksi Menekan Tombol Submit
Hasil yang diharapkan	Sistem akan menampilkan pesan error Bahan tidak tersedia
Hasil	Sistem menampilkan pesan error Bahan tidak tersedia
Status	Valid

6.2.29 Pengujian Update Status Pesanan

Pengujian validasi Update Status Pesanan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_2900. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Update Status Pesanan akan dijelaskan pada tabel 6.51.

Tabel 6.51 Kasus Uji Update Status Pesanan

Nama Kasus Uji	Kasus Uji Update Status Pesanan
Prosedur	1. User masuk kedalam sistem sebagai Kasir 2. Sistem menampilkan halaman utama Kasir 3. Kasir memilih menu Daftar Pesanan 4. Sistem Menampilkan halaman daftar pesanan 5. Kasir menekan tombol Selesai pada pesanan dengan status Kirim ke kasir yaitu dengan Nama Pemesan = "Fakhruzzahid"
Hasil yang diharapkan	Pesanan akan berganti status menjadi selesai
Hasil	Pesanan berganti status menjadi selesai
Status	Valid

6.2.30 Pengujian Mencetak Nota

Pengujian validasi Mencetak Nota dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_3000. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Mencetak Nota akan dijelaskan pada tabel 6.52.

Tabel 6.52 Kasus Uji Update Mencetak Nota

Nama Kasus Uji	Kasus Uji Mencetak Nota
-----------------------	-------------------------

Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Kasir 2. Sistem menampilkan halaman utama Kasir 3. Kasir memilih menu Daftar Pesanan 4. Sistem Menampilkan halaman daftar pesanan 5. Kasir menekan tombol Nota pada salah satu pesanan yaitu dengan Nama Pemesan "Fakhruzzahid".
Hasil yang diharapkan	Sistem akan menampilkan halaman cetak Nota
Hasil	Sistem menampilkan halaman cetak Nota
Status	Valid

6.2.31 Pengujian Membuat Catatan Pesanan

Pengujian validasi Membuat Catatan Pesanan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_3100. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Membuat Catatan Pesanan
Kasus uji Berhasil Membuat Catatan Pesanan akan dijelaskan pada tabel 6.53.

Tabel 6.53 Kasus Uji Berhasil Membuat Catatan Pesanan

Nama Kasus Uji	Kasus Uji Berhasil Membuat Catatan Pesanan
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Konveksi 2. Konveksi memilih menu Jadwal Produksi 3. Sistem menampilkan Daftar Jadwal Produksi 4. Konveksi memilih salah satu Jadwal Produksi yaitu dengan tanggal produksi = "07/08/2018" dan menekan tombol pesanan 5. Sistem Menampilkan halaman List Pesanan 6. Konveksi memilih salah satu pesanan yang status pesanan "Diantrikan" yaitu dengan Nama Pemesan = "Fakhruzzahid" dan menekan tombol detail 7. Sistem menampilkan detail pesanan 8. Konveksi memilih tombol Tambah Catatan 9. Sistem menampilkan form Tambah Catatan 10. Konveksi mengisi form Tambah Catatan dengan data Nama Bahan = "Kain Hitam", Jumlah = "4", Catatan = "-" dan Tanggal Catatan = "07/08/2019". 11. Konveksi Menekan Tombol Submit
Hasil yang diharapkan	Sistem akan menyimpan data catatan pesanan kedalam database dan menampilkan daftar catatan pesanan

Hasil	Data catatan pesanan telah disimpan kedalam database dan menampilkan daftar catatan pesanan
Status	Valid

- b. Kasus uji Form Catatan Pesanan Kosong
Kasus uji Form Catatan Pesanan Kosong akan dijelaskan pada tabel 6.54.

Tabel 6.54 Kasus Uji Form Catatan Pesanan Kosong

Nama Kasus Uji	Kasus Uji Form Catatan Pesanan Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Konveksi 2. Konveksi memilih menu Jadwal Produksi 3. Sistem menampilkan Daftar Jadwal Produksi 4. Konveksi memilih salah satu Jadwal Produksi yaitu dengan tanggal produksi = "07/08/2018" dan menekan tombol pesanan 5. Sistem Menampilkan halaman List Pesanan 6. Konveksi memilih salah satu pesanan yang status pesanan "Diantriikan" yaitu dengan Nama Pemesan = "Fakhruzzahid" dan menekan tombol detail 7. Sistem menampilkan detail pesanan 8. Konveksi memilih tombol Tambah Catatan 9. Sistem menampilkan form Tambah Catatan 10. Konveksi tidak mengisi salah satu inputan yaitu Nama Bahan 11. Konveksi Menekan Tombol Submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form Catatan Pesanan tidak boleh kosong
Hasil	System menampilkan pesan form Catatan Pesanan tidak boleh kososng
Status	Valid

6.2.32 Pengujian Melihat Catatan

Pengujian validasi Melihat Catatan dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_3200. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Catatan akan dijelaskan pada tabel 6.55.

Tabel 6.55 Kasus Uji Melihat Catatan

Nama Kasus Uji	Kasus Uji Melihat Catatan
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Konveksi 2. Sistem menampilkan halaman utama Konveksi

	3. Konveksi memilih menu Daftar Catatan
Hasil yang diharapkan	Sistem akan menampilkan semua daftar Catatan yang telah diinputkan
Hasil	Sistem menampilkan semua daftar Catatan yang telah diinputkan
Status	Valid

6.2.33 Pengujian Menambahkan User

Pengujian validasi Menambahkan User dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_3300. Berikut ini adalah prosedur kasus uji yang dilakukan.

a. Kasus uji Berhasil Menambahkan User

Kasus uji Berhasil Menambahkan User akan dijelaskan pada tabel 6.56.

Tabel 6.56 Kasus Uji Berhasil Menambahkan User

Nama Kasus Uji	Kasus Uji Berhasil Menambahkan User
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Admin 2. Admin memilih menu Tambah User 3. Sistem menampilkan form User Baru 4. Admin mengisi form User Baru yang sesuai yaitu Nama User = "Kasir", Username = "Kasir 2", Password = "Malang" 5. Admin menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data User kedalam database dan menampilkan kembali ke daftar User
Hasil	Data User telah disimpan dan menampilkan halaman daftar User yang sudah diperbaharui
Status	Valid

b. Kasus uji Form User Baru Kosong

Kasus uji Form User Baru Kosong akan dijelaskan pada tabel 6.57.

Tabel 6.57 Kasus Uji Form User Baru Kosong

Nama Kasus Uji	Kasus Uji Form User Baru Kosong
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Admin 2. Admin memilih menu Tambah User 3. Sistem menampilkan halaman form User Baru 4. Admin tidak mengisi form User baru yaitu Username 5. Admin menekan tombol submit

Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form User baru tidak boleh kosong
Hasil	Sistem menampilkan pesan form User baru tidak boleh kosong
Status	Valid

6.2.34 Pengujian Melihat Daftar User

Pengujian validasi Melihat Daftar User dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_3400. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Melihat Daftar User akan dijelaskan pada tabel 6.58.

Tabel 6.58 Kasus Uji Melihat Daftar User

Nama Kasus Uji	Kasus Uji Melihat Daftar User
Prosedur	<ol style="list-style-type: none"> 1. User masuk kedalam sistem sebagai Admin 2. Sistem menampilkan halaman utama Admin 3. Gudang memilih menu Daftar User
Hasil yang diharapkan	Sistem akan menampilkan semua daftar user yang sudah berada pada database
Hasil	Sistem menampilkan semua daftar user yang sudah berada pada database
Status	Valid

6.2.35 Pengujian Mengubah User

Pengujian validasi Mengubah User dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_3500. Berikut ini adalah prosedur kasus uji yang dilakukan.

- a. Kasus uji Berhasil Mengubah User

Kasus uji Berhasil Mengubah User akan dijelaskan pada tabel 6.59.

Tabel 6.59 Kasus Uji Berhasil Mengubah User

Nama Kasus Uji	Kasus Uji Berhasil Mengubah User
Prosedur	<ol style="list-style-type: none"> 1. User masuk sebagai Admin 2. Admin memilih Daftar User 3. Sistem menampilkan Daftar User 4. Admin memilih salah satu Daftar User dengan menekan tombol edit yaitu dengan Nama User = "Kasir2" 5. Sistem menampilkan detail User yang dipilih

	6. Admin mengganti data form User yaitu Nama User = "Kasir", Username = "Kasir 2", Password Baru = "Malang2" 7. Admin menekan tombol submit
Hasil yang diharapkan	Sistem akan menyimpan data User terbaru kedalam database dan menampilkan kembali ke daftar User
Hasil	Data bahan baku terbaru telah disimpan dan menampilkan halaman daftar User yang sudah diperbaharui
Status	Valid

- b. Kasus uji Form User Kosong
 Kasus uji Form User Kosong akan dijelaskan pada tabel 6.60.

Tabel 6.60 Kasus Uji Form User Kosong

Nama Kasus Uji	Kasus Uji Form User Kosong
Prosedur	1. User masuk sebagai Admin 2. Kasir memilih Daftar User 3. Sistem menampilkan Daftar User 4. Admin memilih salah satu Daftar User dengan menekan tombol edit 5. Sistem menampilkan form User yang dipilih 6. Salah satu kolom input kosong yaitu Nama User 7. Admin menekan tombol submit
Hasil yang diharapkan	Sistem akan menampilkan pesan bahwa form User tidak boleh kosong
Hasil	System menampilkan pesan form User tidak boleh kosong
Status	Valid

6.2.36 Pengujian Berhasil Menghapus User

Pengujian validasi Berhasil Menghapus User dilakukan untuk menguji kebutuhan fungsional dengan kode MKB_01_3600. Berikut ini adalah prosedur kasus uji yang dilakukan. Kasus uji Berhasil Menghapus User akan dijelaskan pada tabel 6.61.

Tabel 6.61 Kasus Uji Berhasil Menghapus User

Nama Kasus Uji	Kasus Uji Berhasil Menghapus User
Prosedur	1. User masuk kedalam sistem sebagai Admin 2. Sistem menampilkan halaman Daftar User 3. Admin menekan tombol hapus pada salah satu daftar User yaitu dengan Nama User = "Kasir 2"

	4. Sistem memberi pesan peringatan 5. Admin menekan tombol ok
Hasil yang diharapkan	Sistem akan menghapus User dari database
Hasil	Sistem menghapus User dari database
Status	Valid

6.3 Pengujian Non Fungsional

Pengujian Non Fungsional dari sistem Manajemen Bahan baku dan Pemesanan Sablon malang adalah Compability yang di uji memakai *tools Cross Browser Testing* (crossbrowsertesting.com). Indikator yang diambil adalah *Font size validation, Page Content Alignment, All images and alignment, Header and footer sections*. Untuk perambahan yang dipakai adalah Google Chrome, Mozilla Firefox, dan Opera.

No	Indikator	Google Chrome 65	Mozilla Firefox 61	Opera 51
1	<i>Font size validation</i> (Ukuran font sesuai dan tidak berubah)	Valid	Valid	Valid
2	<i>Page Content Alignment</i> (Penempatan dan penampilan konten dihalaman tetap)	Valid	Valid	Valid
3	<i>All images and alignment</i> (Semua gambar dapat ditampilkan dan ukuran gambar sama)	Valid	Valid	Valid
4	<i>Header and footer sections</i> (tampilan dan tata letak header dan footer sesuai)	Valid	Valid	Valid

6.4 Pembahasan Hasil Pengujian

Hasil dari pengujian unit dan validasi adalah 100 persen valid. Hasil dari pengujian ini menyatakan sistem layak untuk digunakan untuk mengelola data bahan baku dan pemesanan. Pengujian validasi Update Pesanan menghasilkan hasil yang valid dan membuktikan bahwa sistem ini dapat membantu memonitoring pesanan. Pengujian validasi mengecek ketersediaan dan menambahkan catatan menghasilkan hasil yang valid, hal itu membuktikan bahwa sistem ini dapat membantu mengelola bahan baku. Kebutuhan membuat jadwal induk produksi yang telah diuji dengan hasil valid, sehingga terbukti bahwa sistem ini dapat membantu memprediksi proses produksi pesanan. pengujian non fungsional dilakukan dengan berbagai perambanan dan hasilnya 100% valid dan bisa dibuka di berbagai perambanan sesuai indikator.

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil rekayasa Kebutuhan, perancangan, implementasi dan pengujian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut :

1. Berdasarkan hasil rekayasa Kebutuhan, sistem manajemen bahan baku dan pemesanan sablon malang memiliki 36 kebutuhan fungsional dan 6 aktor yang didapat dari hasil observasi dan wawancara. Kebutuhan yang dapat menyelesaikan permasalahan monitoring pesanan, mengelola persediaan dan bahan, mencatat data secara online, merancang jadwal produksi, dan lain sebagainya sesuai Kebutuhan fungsional yang ada.
2. Berdasarkan pada rancangan OO yang telah dilakukan, perancangan menghasilkan perancangan Basis Data dengan membuat PDM dan CDM yang memiliki 11 tabel. Perancangan antarmuka dengan membuat tampilan sementara website. Perancangan arsitektur dengan model *class diagram* dan *sequence diagram*. Perancangan komponen dengan membuat pseudocode untuk mempermudah dalam membaca kode program. Berdasarkan hasil implementasi yang dilakukan sistem ini telah menghasilkan fitur yang dapat membantu mengelola bahan baku, pesanan, jadwal produksi, pembelian bahan baku, memonitoring pesanan, mengantisipasi kehabisan bahan baku saat proses produksi dan pencatatan secara online atau digital.
3. Berdasarkan hasil pengujian yang telah dilakukan dengan menggunakan teknik *white box testing* yang digunakan pada pengujian unit didapat 3 sampai 8 *Cyclomatic Complexity* yang berarti kode terstruktur dengan baik, testability tinggi, serta biaya dan kesulitan rendah, pengujian unit menghasilkan jumlah valid 100%. Begitu juga dengan teknik *black box testing* yang dilakukan pada pengujian validasi pada keseluruhan Kebutuhan fungsional menghasilkan 100% valid. Hal tersebut membuktikan bahwa sistem ini dapat berjalan dan digunakan untuk melakukan manajemen bahan baku dan pemesanan data di sablon malang. Pada pengujian Non Fungsional didapat bahwa disetiap perambahan didapatkan 100% valid berdasarkan indikator yang telah ditentukan.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan sistem manajemen bahan baku dan pemesanan pada sablon malang selanjutnya antara lain

1. Menambahkan fitur penghitungan MPS pada jumlah bahan baku sesuai dengan rumus MPS, karena untuk sekarang MPS masih berupa perkiraan.
2. Memberikan Hak akses dan fitur untuk pemesan agar bisa memonitoring secara langsung status pesanan.

3. Penambahan fitur *Bill Of Material* dengan menyepakati antara sablon malang dan penulis agar bisa dibuat dalam data pasti.



DAFTAR PUSTAKA

- Ambler, S. W., 2012. UML 2 Class Diagrams: *An Agile Introduction*. *Agile Modeling*, [online] Tersedia di: <<http://agilemodeling.com/artifacts/classDiagram.htm>> [Diakses 11 September 2017].
- Arief, M.Rudianto. 2011. *Pemrograman Web Dinamis Menggunakan Php dan Mysql*. Yogyakarta: ANDI.
- Bandur, Agustinus. 2006. "PENELITIAN KUALITATIF Metodologi, Desain, dan Teknik Analisis Data dengan NVIVO 11 Plus". Jakarta : Mitra Wacana Media.
- Blanco, J.A. and Upton, D., 2009, *Codeigniter 1.7*, Packt Publishing, Birmingham.
- Daqiqil, I., 2011. *Framework CodeIgniter Sebuah Panduan dan best Practice*. Pekanbaru: CodeIgniter Lover Community.
- Fowler, Martin. 2005. *UML Distilled Edisi 3*, Yogyakarta: Andi.
- Guru99, 2014. *Learn McCabe's Cyclomatic Complexity with Example*. [online] tersedia di : <https://www.guru99.com/cyclomatic-complexity.html> [Diakses 8 Juli 2018].
- Heizer dan Render. 2014. *Manajemen Operasi*. Jakarta: Salemba Empat.
- Herjanto, Eddy, 2008. *Manajemen Operasi Edisi Ketiga*, Jakarta: Grasindo.
- Indonesia. Kementrian Perindustrian Republik Indonesia. (n.d.). *Peran Ekspor Kelompok Hasil Industri Tekstil Terhadap Total Ekspor Hasil Industri*. Retrieved September 6, 2012, from <http://www.kemenperin.go.id/statistik/peran.php?ekspor=1> .
- Lethbridge, T. & Laganiere, R., 2004. *Object Oriented Software Engineering: Pratical Software Development Using UML and Java*. United Kingdom: Mc Graw Hill-Education.
- Masiyal Kholmi, 2003 "Akuntansi Biaya", Edisi Empat, Yogyakarta, BPFE.
- Page-Jones, M., 2002. *Fundamentals of Object-Oriented Design in UML*. New York: DORSET HOUSE PUBLISHING.
- Pressman, Roger, S. (2001), *Software Engineering: A Practitioner's Approach, Fifth Ed*. New York, McGraw-Hill Book Company.
- Rangkuti, Freddy. (2007). *Manajemen Persediaan: Aplikasi di bidang bisnis*. Jakarta: Raja Grafindo Persada.
- Singgih Wibowo, 2007 "Manajemen Produksi ", Edisi Empat, Yogyakarta, BPFE.
- Sommerville, Ian. 2011. *Software Engineering (Rekayasa Perangkat Lunak)*. Jakarta: Erlangga.
- Stevenson, W. J., 2004, *Operation Management*, 7th ed, McGraww-Hill, USA.
- Wardana, 2010. *Menjadi MasterPHP dengan Framework CodeIgniter*. [e-book] Jakarta : Elex Media Komputindo. Tersedia melalui : Google Books <<http://books.google.co.id>> [Diakses 2 januari 2018].